Students' approaches to mathematical tasks using software as a black-box, glass-box or open-box

Thesis

STUDENTS' APPROACHES TO MATHEMATICAL TASKS USING SOFTWARE AS A BLACK-BOX, GLASS-BOX OR OPEN-BOX

Submitted for the Degree of
Doctor of Philosophy in Educational Technology

Anesa Hosein
16 December, 2009

Institute of Educational Technology
The Open University
Milton Keynes, UK

ABSTRACT

Three mathematical software modes are investigated in this thesis: black-box software showing no mathematical steps; glass-box software showing the intermediate mathematical steps; and open-box software showing and allowing interaction at the intermediate mathematical steps. The glass-box and open-box software modes are often recommended over the black-box software to help understanding but there is limited research comparing all three. This research investigated students' performance and their approaches to solving three mathematical task types when assigned to the software boxes.

Three approaches that students may undertake when solving the tasks were investigated: students' processing levels, their software exploration and their self-explanations. The effect of mathematics confidence on students' approaches and performance was also considered.

Thirty-eight students were randomly assigned to one of the software boxes in an experimental design where all audio and video data were collected via a web-conference remote observation method. The students were asked to think-aloud whilst they solved three task types. The three task types were classified based on the level of conceptual and procedural knowledge needed for solving: mechanical tasks required procedural knowledge, interpretive tasks required conceptual knowledge; and constructive tasks used both conceptual and procedural knowledge.

The results indicated that the relationship between students' approaches and performance varied with the software box. Students using the black-box software explored more for the constructive tasks than the students in the glass-box and open-box software. These black-box software students also performed better on the constructive tasks, particularly those with higher mathematics confidence. The open-box software

appeared to encourage more mathematical explanations whilst the glass-box software encouraged more real-life explanations.

Mathematically confident students were best able to appropriate the black-box software for their conceptual understanding. The glass-box software or open-box software appeared to be useful for helping students with procedural understanding and familiarity with mathematical terms.

DEDICATION

For my grandparents

Tajmul Hosein

Phulbassie Bocas (1912-2009)

Assia Hosein (1916-2008)

Salim Bocas (1912-1993)

Table of Contents

List of Tables

List of Figures

List of Appendices

Acknowledgements

My first acknowledgement is to my three supervisors, James, Doug and John, who have helped and supported me immensely. I have enjoyed the road to becoming an independent researcher with you guys; thank you for guiding me on the correct path and steering me away from potential pitfalls. I cannot imagine doing this without you! Also thanks to my internal (Patrick McAndrew) and external (Sara Hennessy) examiners for their comments on my thesis.

To my three gatekeepers, Amalia, Ritchie and Colin, I give you my eternal gratitude. I can honestly say without your help this research would never have happened, thank you for taking your time and energy to help me get my participants. A special thanks to all those students in Trinidad and Tobago who took part in my survey, even though the remuneration was in chocolate bars.

I want to thank Donna Phillips and Emma Street who helped me immensely in sending off the online surveys for my Support Study and who did not complain (well, not loudly) no matter how many items I wanted to change or correct.

Gill and Rebecca, this has been a long but interesting journey as we reached our goal for our PhDs – I am glad you were there along the way to support and share information about the joys and trials of academia. Also, thanks to the numerous persons who provided feedback on my presentations and publications (see Appendix 8, p.333).

Thanks to Tina, Nana and Jenny and my various housemates who listened to all my rants about my PhD and my hiccups in student visas, even though half the time you had no clue what I was speaking about. An extra special appreciation to my family: they were never-ending in their support for me emotionally (and sometimes financially!) in my time here in the UK whilst I follow my dream. Finally, I am grateful through the Grace of God, I have been able to reach this far.

Chapter 1. Introduction

"Learn not only to find what you like, learn to like what you find."
- Anthony J. D'Angelo

1.1 Chapter Outline

This chapter begins with the research background (Section 1.2) and the rationale for this research (Section 1.3). In the rationale, relevant literature is highlighted from which preliminary research questions are outlined (Section 1.4).The chapter finishes with an outline for the rest of the thesis (Section 1.5).

1.2 Research Background

Various types of mathematical software are frequently employed in the teaching and learning of mathematics at various levels of education (primary, secondary and tertiary). Mathematical software may be used either in advanced or graphical calculators, or on a computer. Software packages can be a) generic mathematics packages such as computer algebra systems (CAS), b) ubiquitous software such as spreadsheets or c) dedicated software for specific mathematical topics such as Lindo for linear programming (see Crowe and Zand, 2001; Dana-Picard and Steiner, 2004; Gass, Hirshfeld and Wasil, 2000; Hosein, 2005).

Most mathematical software packages could probably be classified as what Buchberger (1990; 2002) termed 'black-box'. In black-box software, tasks are solved without showing the intermediate steps. Traditionally, ubiquitous software such as spreadsheets, generic mathematical software and dedicated software have all been used as black-box software. However, recent software packages such as that employed by the Casio FX 2.0 calculator are able to show the steps during the solution process in its tutorial mode (Horton, Storm and Leonard, 2004). Buchberger referred to this mode of mathematical software that shows the internal steps of the algorithms as 'white-box' software. The terms, black-box and white-box, were derived from Myers (1979), who used it in reference to software testing. In the software engineering testing literature, white-box is also referred to as glass-box (e.g. Bache and Mullerberg, 1990) and this is the terminology that is used in this thesis.

Dana-Picard and Steiner (2004) explained that traditional black-box software can be used as more than ‘number crunchers’ for mathematical tasks, if ‘low-level’ commands are utilized by the students. Low-level commands are where each of the mathematical steps in a task is solved separately until the solution is achieved, thus enabling the students to see the solution procedure. Software packages are also produced which can mimic the solving of each step that Dana-Picard and Steiner have proposed. These software packages allow the student to examine and interact at the intermediate steps of the task. This software mode is different from the black-box and the glass-box and is classified separately. In this thesis, they are referred to as open-box software as the students are allowed to look into the software and use it. Open-box software has not been classified separately in any literature before.

An illustration of these software boxes is shown in Figure 1. The bubble on top of each box shows the equation that the students input for solving the variable, x. The bubble at the bottom of each box shows the value of x computed by the software box. With the black-box software, the students are unable to see the steps for solving x, whilst in the glass-box software the students are able to see the steps. In the open-box software, the lines, on the right-hand-side connected to each step, show the additional input by student for solving the task.

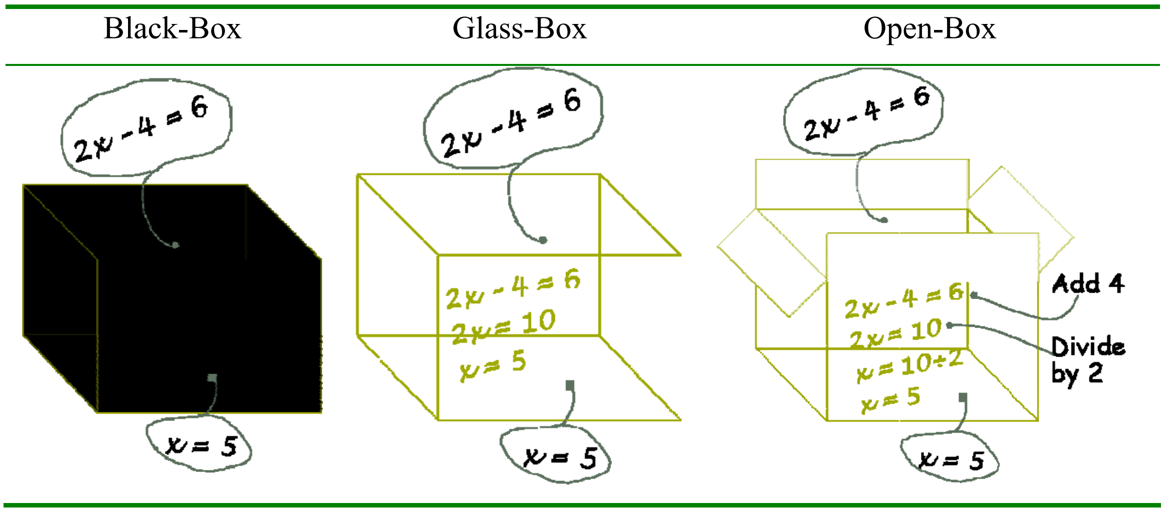


Figure 1: An illustrative comparison of an algebra solution by black-box, glass-box and open-box software

The aim of this thesis is to provide insight on how these three software boxes influence the understanding of mathematics at the university level. Past research studies investigating mathematical understanding utilising software have relied mainly on the black-box software; this thesis will extend the research into glass-box and open-box software.

1.3 Rationale

There are mixed views on the use of software in the understanding of mathematics. The supporting views often pertain to black-box software (including typical calculators) as this is the type commonly used in schools and universities (Crowe and Zand, 2000; Dreyfus, 1994). On one hand, black-box software packages are viewed positively because they enable students to reduce the quantity of tedious computations, lessen their anxiety about performing computations accurately and they allow students to work with more realistic examples (Heid and Edwards, 2001; Whiteman and Nygren, 2000).

On the other hand, black-box software packages are seen as potentially detracting from conceptual understanding by permitting students to solve problems by trial and error and encouraging students to accept calculated answers unquestioningly

(Buchberger, 1990; Crowe and Zand, 2000; Dana-Picard and Steiner, 2004; Whiteman and Nygren, 2000). Although there is a suggestion that the black-box software hampers conceptual understanding, studies on mathematical learning with black-box software (usually using a CAS or graphical calculator) have suggested the opposite. These studies implied that the black-box software promoted more of the students' conceptual understanding when compared to those students who learnt without any software (see Heid, 1988; O'Callaghan, 1998; Palmiter, 1991).

Conceptual understanding is related to conceptual knowledge and the latter is defined by Rittle-Johnson, Siegler and Alibali (2001) as the "implicit or explicit understanding of the principles that govern a domain and of the interrelations between units of knowledge in a domain" (p.346-347). There is also the associated concept of procedural knowledge which Rittle-Johnson *et al.* (2001) defined as "the ability to execute action sequences to solve problems" (p.346). Thus, conceptual understanding is the construction of relationships between information which helps students to further their understanding, whilst procedural learning is where students practise following defined steps to solve tasks. Both of these are discussed further in Section 2.2 (p.15).

As already mentioned, the study by Heid (1988) found that students who used black-box software (i.e. CAS) gained greater understanding than those students who used no software. However, this perceived increase in conceptual understanding was probably not due entirely to the black-box software. In her study, the students using black-box software were exposed to pedagogical methods that could improve conceptual understanding. The students using the black-box software were taught to think more conceptually by the teacher since the teacher's pedagogical emphasis was on concepts. For the students who did not have the black-box software the emphasis was on skills and algorithms. Thus, the difference that Heid found between these groups probably was because of how the mathematical topic was taught. Similar discrepancies in how concepts were presented to and compared between groups were noted in separate studies by O'Callaghan and Palmiter (see Appendix 1, p.282). Therefore, there is still uncertainty as to whether the use of black-box software can improve the conceptual understanding of students.

Buchberger (1990) pointed out that students who were taught using black-box software were unable to gain mathematical insight and learn general solving techniques. Further, Star and Seifert (2006) highlighted that students who were privy to multiple solution procedures and able to invent their own procedures were more likely to have a greater conceptual understanding. Therefore, as suggested by Buchberger, the glass-box software (shows the intermediate steps) can provide a way for students to investigate solution procedures and even invent their own. The open-box software is probably better poised than the glass-box software to allow students' investigation of solution procedures. This is because the open-box software can allow students to interact and also investigate different solution steps. 'Interact' here means that the student would have to perform a manoeuvre before the next step is shown, for example, adding or subtracting values in an equation.

However, there are few research studies into how students' use of glass-box and open-box software may promote their understanding. Strickland and Al-Jumeily (1999) in their study of 26 secondary school students investigated how one group of students learnt algebra with open-box software (a CAS called TREEFROG) compared with another group using no software. After five 50-minutes sessions, students were given a hand-written test which examined procedural knowledge. Strickland and Al-Jumeily found from the hand-written tests that students using the open-box software performed better in the procedural tasks than those students who did not use any software.

All of the studies discussed above have compared one software mode against not using any software. It, therefore, makes it difficult to ascertain where one software mode may promote better mathematical understanding compared to others. The study by Horton *et al.* (2004) addressed this situation to some extent by comparing two software modes, the glass-box and black-box software. They conducted their study using 50 undergraduate students learning linear equations over a three week period. Through a written examination which tested procedural knowledge, they found that students who used the glass-box software performed better than those who used the black-box software. This opens an opportunity to explore more iterative types of software (that is, glass-box and open-box software), in particular, how the software boxes impact on students' use of procedural and conceptual knowledge for their understanding.

There is thus some evidence to show that students perform better in written examinations after they used the glass-box software when compared to students using the black-box software, but there are no formally documented comparisons using all three software boxes. Further, there is research which suggests that some software boxes promote procedural understanding, but studies relating to whether software boxes can promote any conceptual understanding have been limited and inconclusive.

Thus far, mathematical understanding has been described as consisting of two separate forms of knowledge, that is, conceptual and procedural. Some researchers, however, believe that the two types of knowledge are linked (see Kadijevich and Haapasalo, 2001; Rittle-Johnson *et al.*, 2001; Tall *et al.*, 2001). Rittle-Johnson *et al.* (2001) suggested that conceptual and procedural knowledge lie on the extreme ends of a continuum and that they influence each other in a bi-directional manner: that is, any improvement in a student's conceptual knowledge is linked to an improvement in the student's procedural knowledge and vice versa.

Therefore suggesting that tasks represent purely conceptual or purely procedural knowledge may be flawed. Using Galbraith and Haines (2000a) classification of mathematical tasks is one way of addressing this flaw. They suggested using three task types based on the conceptual and procedural continuum. These task types are mechanical, interpretive and constructive. Mechanical tasks use mainly procedural knowledge, interpretive tasks use mainly conceptual knowledge and constructive tasks use a mixture of both conceptual and procedural knowledge. Table 1 presents examples of all three task types. The description column indicates the methods for solving the tasks. Where conceptual and procedural knowledge are needed for solving the tasks, these are presented in the brackets.

This taxonomy is easily applicable to previously conceived tasks. For example, from examining the tasks in the studies mentioned above, it was noted that the tasks from Horton *et al.* (2004) were all mechanical tasks (representing procedural knowledge) whilst those from Heid (1988) were either interpretive or constructive tasks (representing conceptual knowledge).

Table 1: Examples of Galbraith and Haines (2000a) classification of task types

Task	Description
Mechanical: Find the minimum point of the function $y = (x - 4)^2 + 10$	Differentiate the equation to determine what the minimum point is (procedural)
Interpretive: If the function y is modified to be $y = (x - 4)^2 + 3$, in what direction would the minimum point be shifted?	Recognise, by using the general quadratic equation $y = a(x - h)^2 + k$, that k represents a vertical shift. k has been reduced and thus would mean that the minimum point would be shifted downwards (conceptual)
Constructive: What is the new minimum point of the modified function y and what is the reason why it is different from the original function?	Recognise that the shift is downwards and hence means that the original vertical point would be shifted by 7 units downwards (conceptual) and that the new minimum point coordinates will be $(x_{\text{original}}, y_{\text{original}} - 7)$ (procedural).

In the past, educators have recommended that lecturers should use glass-box or open-box software instead of the black-box software when teaching students. For example, Buchberger (1990; 2002) indicated that students should be taught the solution steps when solving problems otherwise the students may lose important mathematical problem solving skills. He then suggested the use of the glass-box over black-box software for assisting students in understanding the solution steps. These steps perhaps could also be taught through the use of pen-and-paper or by using the open-box software. Buchberger further indicated that once students become practised in performing the procedural steps they then could move on to using the black-box software since the procedural steps are now trivialised. For example, if students solved a series of long division problems by pen-and-paper, then over time, the students would be able to apply the governing procedural steps without hesitation, that is, the steps have become trivialised. At the point where the steps become trivialised, students should then be allowed to use a computer or a calculator for computational purposes.

Winston (1996), on the other hand, recommended that students need only to learn the procedural steps if they are relevant to their discipline. For example, he indicates that for finance and marketing students, it is not necessary to learn how to calculate the steps of the simplex algorithm associated with linear programming as these students would not need this knowledge later in their lives or jobs. However, he suggested that students in the mathematical disciplines should learn how to calculate the steps as they may be able to improve the linear programming solving method.

Whilst this debate is ongoing, this leaves lecturers in a predicament. The lecturers are uncertain of which software box to use in their teaching and whether they are doing a disservice to their students by omitting the procedural steps since this exclusion minimises the development and application of students' mathematical skills. Indeed the debate may even suggest that procedural knowledge should not be taught and students should only concentrate on learning conceptual knowledge. However recent studies (e.g. Rittle-Johnson and Alibali, 1999) have shown that acquiring one type of knowledge can affect the uptake of the other, that is, conceptual knowledge affects the uptake of procedural knowledge and vice versa. Thus omitting to help develop one type of knowledge may affect how students are able to use their other knowledge type effectively.

Thus, the primary concern of this thesis is trying to understand how students' understanding is affected by the three software boxes. However, the research outcomes should also be able to provide recommendations for teaching practices and software development. For example, in teaching, one software box can be recommended as the best for promoted students' use of conceptual or procedural knowledge. The research could however find that students used the three software boxes differently and the software boxes helped develop different aspects of the students' understanding. This finding may then impact on which software boxes should be included in the development of a mathematical software package.

1.4 Research Questions

The studies identified previously including the comparison of students' performance when using the software boxes are presented in Table 2. These studies demonstrate that there is limited research in the measuring of students' mathematical understanding when assigned to all of the three software boxes in the three task types.

In all of these studies, students' performance on tasks was used as an indicator for mathematical understanding. Horton *et al.* (2004) showed that students' performance on mechanical tasks depended on their assigned software boxes, albeit the comparison was only for the black-box and glass-box. Unlike the glass-box software, the open-box software allows a student to see and interact and determine the next step.

These students using the open-box software are thus able to practice and learn the steps associated with mechanical tasks. Thus, in a written examination, students who have used an open-box software may perform better on mechanical tasks than those who have used a glass-box or black-box software.

Table 2: Students' performance on tasks when using the software boxes for selected studies

Software Comparison	Tasks (Examination)	Studies	Findings on students' performance
Black-box versus No Software	Interpretive and Constructive (written test)	Heid (1988), O'Callaghan (1998), Palmiter (1991)	Black-box performed better but may have been influenced by pedagogy
Open-box versus No Software	Mechanical (written test)	Strickland and Al-Jumeily (1999)	Open-box performed better
Black-box versus Glass-Box	Mechanical (written test)	Horton *et al.* (2004)	Glass-box performed better

It is interesting to note that in the study by Horton *et al.*, performance was assessed through a written test when comparing black-box and glass-box software. Whilst this is justifiable for studies comparing with and without software, the reason for a written test for comparing the glass-box and black-box software is unclear. This meant that any approach that students undertook in solving the mechanical task when using the software boxes was lost. Understandably if open-box software was being used, then students using glass-box and black-box software may be at an advantage as calculations will be performed quicker in the glass and black-box software since the students using the open-box software will need to interact at each step.

The other two tasks identified by Galbraith and Haines (2000a) are different from the mechanical tasks as they require mostly conceptual knowledge (interpretive) or a mixture of conceptual and procedural knowledge (constructive). The software boxes are used primarily in solving procedural tasks and thus, similarly to the mechanical tasks, the performance of students on constructive tasks may vary with the software box. However, since conceptual knowledge and procedural knowledge are probably linked (e.g. Rittle-Johnson and Alibali, 1999), they can influence each other. For example, the procedural steps being shown in the glass-box and the open-box software may influence the students' conception of the task and may impact on their performance for interpretive tasks.

Thus, the first research question for this study was:

1. *Does students' performance in solving the three task types (mechanical, interpretive and constructive) depend upon the software box they have access to?*

This research question uses the phrase 'software box they have access to'. This term means that the students have access to a software box when they are solving tasks and not at some later time as in a written examination. For mechanical tasks, students solving this task type with the software boxes should solve it correctly whether using black-box or glass-box software, providing they enter the correct values. For the open-box software there is more room for error in a mechanical task as the students have to decide on the manoeuvre they have to perform.

Galbraith and Haines (2000a) indicated that the interpretive and constructive tasks were more difficult than the mechanical tasks for students to solve, and they found that the students performed worst in the constructive tasks. This trend is expected to be reflected in the task scores (i.e. performance) of students regardless of the software boxes, that is, students should perform worst on the constructive tasks. However, whether students' task scores will be similar for all software boxes is uncertain. If the task scores are similar regardless of the software boxes, there is a possibility that the students' approach in using the software boxes may vary as the software boxes are different. On the other hand, if the students' task scores were different for the software boxes, knowing whether the students' approaches in solving tasks were dissimilar can probably flag potential strengths and weakness of each software box. Thus the second and third research questions for this study were:

2. *Do students' approaches to solving the three task types (mechanical, interpretive and constructive) depend upon the software box they have access to?*

3. *How are students' approaches to solving the three task types (mechanical, interpretive and constructive) associated with their performance? And does this association vary with the software box they have access to?*

Finally, no observed variation may be found in the approaches that students took when solving the tasks for a particular software box. An inquiry into how each software box aided the students in solving the three tasks may be pertinent. As a result, the final research question for this study was:

> *How do students' approaches to the three task types (mechanical, interpretive and constructive) and their performance on these tasks depend on the software box they have access to?*

In fact, this final research question, in itself, was the overarching question with the other three research questions acting as subsidiary questions to answer it. These research questions are again refined in Section 2.10.1 (p.50).

1.5 Thesis Outline

This thesis has 7 chapters. This first chapter has provided an introduction with some background information to the research that was conducted. Chapter 2 expands on the studies presented here in Chapter 1. Further, literature is examined to determine how to measure students' performance and their approaches in solving the tasks. Three approaches are identified: explanations, explorations and processing levels. From this identification of approaches, an analytical framework is developed for comparing the tasks and software boxes. This framework also includes mathematics confidence as an attitudinal variable. The research questions are then refined.

Chapter 3 outlines the methodological choices for this research, in particular how performance and approach to solving tasks are operationalised and measured. This chapter briefly outlines the four studies in this thesis (Supporting Study 1, Pilot Study 2, Pilot Study 3 and the Main Study). The data collecting method of remote observation is also explained as this allowed the recording of audio and video data of the participants via the internet. Chapter 4 discusses the results from Supporting Study 1 and the two pilot studies. The two pilot studies were used to test the remote observation method, the software boxes that were developed for this thesis and the tasks. The results from these studies influenced the design of the Main Study.

The quantitative data (Chapter 5) provides the statistical analysis of differences in performance and approaches amongst the software boxes with respect to the task types. Chapter 6 presents the qualitative illustrations of how students answered the three task types with respect to the software boxes and discusses the influence of approaches. Chapter 7 concludes with the main findings of the studies and suggests recommendations for future research.

Chapter 2. Performance and Approaches

"Knowledge is not a series of self consistent theories that converges towards an ideal view; it is rather an ever increasing ocean of mutually incompatible (and perhaps even incommensurable) alternatives, each single theory, each fairy tale, each myth"
- Paul Feyerabend

2.1 Introduction

This chapter highlights and discusses the relevant literature in this thesis. In Chapter 1, the aim of this thesis was presented as determining the influence of the three software boxes on students' mathematical understanding. In the reported studies highlighted in Chapter 1, students' performance was used as a measure of mathematical understanding. This was operationalised by measuring students' performance on tasks that required either conceptual or procedural knowledge. This chapter, firstly, elaborates on these knowledge types as a way of measuring mathematical understanding (Section 2.2, p.15).

This section is followed by discussing how both conceptual and procedural knowledge can be operationalised to determine students' performance (Section 2.3, p.18). This links with the three task types (mechanical, interpretive and constructive) mentioned in Chapter 1. The studies mentioned in Chapter 1 on the software boxes and the measurement of conceptual and procedural knowledge, are then elaborated on. From these studies, inferences are made regarding students' expected performance on the three task types and software boxes to help answer Research Question 1.

Whilst performance can measure the extent of students' mathematical understanding, it is unable to show the pathway or approach undertaken by a student to acquire the solution. The approaches that a student may undertake relate to Research Question 2. Three students' approaches are then identified in the chapter: a) deep/

surface processing level, b) explanations and c) explorations (Section 2.4, p.26). These approaches are not considered exhaustive of all the approaches that a student can embark on and neither are they mutually exclusive to each other.

The chapter then discusses each of these approaches in terms of how they may influence performance and how they relate to each other (Sections 2.5, 2.6 and 2.7). Inferences are also made on what approach students may take depending on which software box they have access to. These inferences should help in answering Research Question 3.

To account for attitudinal differences, self-efficacy is discussed for its influence on performance, the approaches and use of software boxes (Section 2.8, p.42). Finally, an analytical framework is presented to possibly show how performance is connected to the approaches and self-efficacy (Section 2.9, p.46). This analytical framework was also used for analysing the qualitative data in Chapter 6. The chapter concludes with the main points made and a refinement of the research questions (Section 2.10, p.49).

2.2 Conceptual and Procedural Knowledge

Both conceptual and procedural knowledge are acquired during learning by students. This section elaborates on conceptual and procedural knowledge and their connection with mathematical understanding. In particular, this section sets the basis for how performance can be an operationalised form of acquired conceptual and procedural knowledge by showing the link between knowledge and understanding.

In Section 1.3 (p.3) definitions of conceptual and procedural knowledge were presented from Rittle-Johnson *et al.* (2001). There have been alternative definitions in various disciplines for both types of knowledge (see for example Anderson, 1995; de Jong and Ferguson-Hessler, 1996). These definitions are all consistent that conceptual knowledge deals with relationships, whilst procedural knowledge deals with steps or rules.

The definitions of procedural and conceptual knowledge most often used within the mathematical domain are from Hiebert and Lefevre (1986). According to Hiebert and Lefevre (1986), conceptual knowledge is “rich in relationships” (p.3) and “cannot be an isolated piece of knowledge” (p.4). Isolated pieces of information can be “part of conceptual knowledge only if the holder recognises its relationship to other pieces of information” (p.4). Further, the formation of conceptual knowledge occurs “between two pieces of information that already have been sorted in memory or between an existing piece of knowledge and one that is newly learned” (p.4).

On the other hand, Hiebert and Lefevre noted that procedural knowledge is discrete knowledge and consists of two parts. The first part is “composed of the formal language, or symbol representation system, of mathematics” (p.5). Secondly, procedural knowledge is concerned with algorithms or rules, in particular the step-by-step instructions for completing a task.

These definitions show that all knowledge acquired cannot be sorted into procedural and conceptual knowledge. Another type of knowledge that is common within the cognitive psychology literature is declarative knowledge which is often distinguished from procedural knowledge. Anderson (1995) defines declarative knowledge as the knowledge of facts: it is explicit knowledge of states of affairs, whereas, procedural knowledge is knowledge of how to do things which are not necessarily explicit.

Through acquiring conceptual and procedural knowledge, understanding may arise. Skemp (1976) classified ‘understanding’ into relational and instrumental. Relational understanding was explained as “knowing both what to do and why” (p.2) whilst instrumental understanding was using “rules without reasons” (p.2). Relational understanding is associated with conceptual knowledge. Conceptual knowledge is about making links with previous knowledge but in relational understanding, the students know how to use their conceptual knowledge for solving a task. Instrumental understanding is associated with procedural knowledge, in that students follow steps and procedural knowledge can be applied with or without reason to tasks. If applied with reason, students are using procedural knowledge for relational understanding but if used without reason, then students are applying procedural knowledge through instrumental understanding.

Associated with conceptual and procedural knowledge are two types of learning: meaningful and rote learning. Meaningful learning, according to Hiebert and Lefevre (1986), is rich in relationships and thus all conceptual knowledge “must be learned meaningfully” (p.8). On the other hand, procedural knowledge may or may not be meaningful. If procedural knowledge was acquired via meaningful learning then according to Hiebert and Lefevre, the procedural knowledge is linked with conceptual knowledge.

The second type of learning, rote learning is when acquired knowledge is absent of relationships as it is “tied closely to the context in which it is learned” (p.8). The knowledge acquired is “accessed and applied only in those contexts that look very much like the original” (p.8) In other literature, particularly psychological experiments, this is referred to as near-transfer skills (e.g. Renkl, 1997). Hiebert and Lefevre also explained that conceptual knowledge cannot be acquired through rote learning, in that “facts and propositions learned by rote are sorted in memory as isolated bits of information, not linked with any conceptual network” (p.8).However at some later stage, the student may “recognise or construct relationships between isolated pieces of information” (p.8) and thus acquire conceptual knowledge from information that was originally learnt through rote learning. In the situation of learning procedural knowledge through rote learning; Skemp's reference to 'rules without reason' may apply and this may reflect instrumental understanding.

In this view, the quality of conceptual and procedural knowledge acquired by a student would depend on their method of learning, that is, either meaningful or rote. For example, if they have meaningful learning of concepts such that they know what to do and why, then they will have acquired good understanding (relational). If however they used rote learning of procedures, then they will have succeeded in acquiring instrumental understanding. Their abilities to tackle tasks will be affected by their gained understanding. Therefore, the extent to which students' conceptual or procedural knowledge aid them in solving tasks would depend on whether their understanding was acquired through meaningful or rote learning.

2.3 Performance

Mathematical understanding is determined by how well students apply their previously gained conceptual or procedural knowledge to tasks. Mathematical understanding can be measured by students' ability to correctly solve tasks that are specifically created to use conceptual or procedural knowledge. The extent to which the students are able to solve these tasks correctly will be reflected in their performance. The use of students' performance as an indicator for mathematical understanding is similar to a summative assessment (Bloom, Hastings and Madaus, 1971).

It could be argued that students using relational understanding should outperform students using instrumental understanding on conceptual-based tasks. Further, for tasks that require the use of conceptual and procedural knowledge, students using relational understanding would again have the advantage and should do better than students using instrumental understanding. For tasks requiring mostly procedural knowledge, then perhaps students using either relational or instrumental understanding should perform similarly.

Therefore, the chosen tasks need to represent a range of understanding and it is important that these tasks are representative of both conceptual and procedural knowledge. This section on ‘Performance’ thus looks at how tasks can be classified into conceptual and procedural knowledge by using the three task types (mechanical, interpretive and constructive) proposed by Galbraith and Haines (2000a) and referred to in Section 1.3 (p.3). This is followed by considering issues in measuring performance with software in general and the issues of measuring performance for the three task types using the software boxes.

2.3.1 The Three Task Types

In this section, a brief history is provided on the development of the mechanical, interpretive and constructive task types by Galbraith and Haines. These three tasks types were developed based on the Mathematical Assessment Task Hierarchy (MATH) taxonomy of Smith *et al.* (1996). The MATH taxonomy was in turn based on Bloom (1956) and was modified for mathematics. Smith *et al.* grouped mathematical tasks into three types, A, B and C (see Table 3) based on increasing difficulty where A was the easiest and C the hardest.

Table 3: The MATH taxonomy

A	B	C
Factual Knowledge	Information transfer	Justifying and interpreting
Comprehension	Application in new situations	Implications, conjectures and comparisons
Routine use of procedure		Evaluation

Galbraith and Haines suggested that the taxonomy represented the extent of conceptual and procedural knowledge required for solving the tasks. Hence, students would perform the best in group A tasks, which Galbraith and Haines termed ‘mechanical’ tasks, as these tasks required only procedural knowledge, and worst in the Group C tasks (‘constructive’ tasks) as these required both a mixture of conceptual and procedural knowledge. Galbraith and Haines thus labelled the groups of tasks as mechanical (A), interpretive (B) and constructive (C). Through written exams over a period of three years with 423 university students studying graphical representations in algebra, they confirmed their hypothesis. Students consistently performed the best in the mechanical tasks, the worst in the constructive tasks and intermediate in the interpretive tasks.

Rittle-Johnson *et al.* (2001) argued that conceptual and procedural knowledge lie on the ends of a spectrum. If tasks were developed to use only conceptual or only procedural knowledge, then using the results from Galbraith and Haines, this will suggest that the easiest tasks for the students are towards the procedural side and the hardest tasks are towards the conceptual side of the spectrum. Using the spectrum of conceptual and procedural tasks will suggest that tasks requiring a mixture of conceptual and procedural knowledge should be easier than tasks needing purely conceptual knowledge. However from Galbraith and Haines, this is known not to be true, as constructive tasks which required both conceptual and procedural knowledge were the hardest. Perhaps then conceptual and procedural knowledge do not lie on a spectrum, but rather the conceptual-procedural knowledge is a hybrid knowledge requiring the linking and mastery of both knowledge types. This ability of students, to apply a mixture of conceptual and procedural knowledge, could be argued to point to a superior mathematical understanding compared with those students who can only apply purely conceptual or purely procedural knowledge.

Therefore, students' level of understanding can be inferred from their performance in the three task types. Students who have a better mathematical understanding can be identified through a higher score in the constructive tasks.

2.3.2 Challenges of Measuring Performance when using Software

In the previous section, the task types were defined to ensure that the measured performance operationalised the students' use of conceptual or procedural knowledge. The focus in this section is looking at the challenges of measuring performance when software is involved.

One of the difficulties with using mathematical software is knowing whether the measured performance is a true indicator of the students' ability to follow software instructions on how to solve the tasks or whether the students have genuinely applied their mathematical understanding for solving the tasks whilst using the software as a tool (Buchberger, 2002; Meagher, 2000; Tall, 1994). The first could be likened to instrumental understanding, where steps are followed without reason, and the second to relational understanding, where the students know not only what to input into the software but why the calculation is necessary. Students knowing 'when' to use software could therefore be seen as an additional aspect of relational understanding.

2.3.3 Expected Performance using the Task Types and Software Boxes

As noted in Section 1.3 (p.3), research studies into measuring performance when students are using the glass-box and open-box software have been limited. The main concern of the known research studies was determining whether these software boxes (glass-box or open-box) aided in students' procedural knowledge when compared to students using the black-box software or pen-and-paper. In the study by Horton *et al.* (2004) they found that students who were trained with the glass-box software versus black-box software outperformed their counterparts via a written test. Similarly, Strickland and Al-Jumeily (1999) found that the students who were trained with the open-box software versus those students who learnt with pen-and-paper outperformed their counterparts in procedural knowledge via a written test (see Section 1.2, p.3). This is not a completely surprising result considering that, in both the glass-box and open-box software, students are presented with or trained to understand the steps, which is procedural knowledge. However, there are no studies indicating whether either of these software boxes (glass-box and open-box) may help in conceptual understanding.

Interestingly, in the studies involving only the black-box software, the main focus was determining students' performance on conceptual tasks by comparing student's scores using the software versus a pen-and-paper method. These studies with the black-box software included that of Heid (1988), Palmiter (1991) and O'Callaghan (1998) (more information on these studies are provided in Appendix 1, p.282, see also Section 1.3, p.3). The conceptual tasks used in these studies can be grouped either into interpretive and constructive (see Figure 2 for an interpretive task). All three studies showed that students performed better when taught with the black-box software than when taught without. Although inferences made from these studies will be most relevant to black-box software, they will also be extended to the glass-box and open-box software.

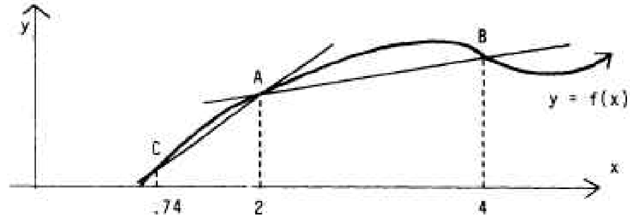
Question	Source	Experimental section 1	Experimental section 2	Comparison section
1. You are given the graph below and the following two facts: The line containing A and B has slope equal to 1/2. The line containing A and C has slope equal to 2.  Which, if any, of the following statements must be true? For each of the statements which is true, explain why. (You may wish to name more than one statement.)	Quiz 1			
a. The slope of the curve y = f(x) at x = 4 is greater than 1/2.		90	91	84
b. The slope of the curve y = f(x) at x = 2 is less than 2 and greater than 1/2.		85	86	73
c. The slope of the curve y = f(x) at x = .74 is greater than 2.		75	68	59

Figure 2: An interpretive (conceptual) task from Heid (1988) showing the scores obtained by the non-software (Comparison) and software (Experimental) students

Dreyfus (1994) suggested that through the use of software (black-box), students were able to do the ‘trivial computations’ such as finding task solutions. This enables the students to “operate at a high conceptual level … they can concentrate on the operations that are intended to be the focus of attention and leave the lower-level operations to the computer” (p.205). Perhaps then in these three studies, the black-box software reduced the cognitive effort required for computation and the students were able to concentrate their cognitive effort towards applying their conceptual knowledge. However, Crowe and Zand (2000) noted that whilst computer software can carry out the trivial computations; the software does not replace conceptual thinking.

From the three studies (that is, the studies by Heid , Palmiter and O'Callaghan), it was difficult to ascertain whether the students were free to concentrate on using their conceptual knowledge. The main reason for this difficulty was that the students using the black-box were taught to think conceptually by looking at multiple representations. The phenomenon of students doing better in conceptual tasks when using the black-box software can probably be attributed to the teacher: specifically, the teaching method 'privileged' the use of conceptual knowledge (Kendal and Stacey, 1999). Appendix 1 (p.282) has more information on this study.

As Kendal and Stacey (1999) noted, when students use mathematical software such as computer algebra systems (CAS), although conceptual errors in tasks are not eliminated, procedural errors such as incorrect arithmetic can be. This suggests that mechanical problems should always be solved if students decide to use mathematical software as procedural errors are minimised, providing that the correct numbers are inputted or they have chosen the correct procedure (such as integration, differentiation etc. in the software). Tasks completed through software compared to pen-and-paper should have little or no procedural errors, as these errors are eliminated through the use of software.

Kendal and Stacey were referring to black-box software, and perhaps this argument can extend to glass-box software as well, since the only difference between both software boxes is that the steps are shown in the latter. However, in the open-box software where students have to decide and interact at each step, then procedural errors may still occur. Thus, students using black-box or glass-box software should obtain high performance scores because of their ability to perform the algorithm in mechanical tasks without making procedural errors.

Interpretive tasks require mostly the use of conceptual knowledge. As little procedural knowledge is needed for the task, the probability of students using any of the software boxes to solve this task may be quite low. This does not mean the software boxes cannot impact on the performance of the interpretive task. As noted in the previous Section 2.3.1 (p.19), conceptual and procedural knowledge may be linked. Rittle-Johnson and Alibali (1999) showed that to some limited extent an improvement in conceptual knowledge could lead to an improvement in procedural knowledge and vice versa. Drijvers (2000) found that some students like to know what is occurring and perhaps by showing these students the procedural steps, they will then try to make sense of the steps. Students who, therefore, used either the glass-box or the open-box software in solving mechanical tasks would be privy to the steps (that is, the procedural knowledge). Students who try to make sense of the steps may then improve their conceptual knowledge of the mathematical topic, that is, acquire a relational understanding. However, if the students are only willing to apply the rules without making sense of the steps, they may only gain an instrumental understanding.

Therefore, there is a possibility that students having access to the open-box and glass-box software may outperform students using the black-box software in the interpretive tasks, providing the students, with the glass-box and open-box software, spent time trying to understand the steps conceptually. Perhaps as the students in the open-box software are 'forced' to do the steps, these students may gain the most conceptual knowledge from using this software and then outperform even the glass-box students in the interpretive tasks. This is all with the proviso that students actually engage with the open-box software and form conceptual knowledge, that is, relevant to the interpretive task.

Constructive tasks require both the use of conceptual and procedural knowledge. For students to be successful at these tasks, they have to recognise the link between the procedural and conceptual aspect of the task. There is a possibility that students using glass-box or open-box software may concentrate on using their procedural knowledge by trying to make sense of the steps but devote little cognitive effort towards conceptual thinking. However, those students using the black-box software will not be sidetracked by the procedural steps and perhaps can devote most of their time towards conceptual thinking. Therefore this argument suggests that students with the black-box software should perform better on the constructive tasks than those with the glass-box and open-box software. It is still quite unclear which software box will produce the best performance on the three tasks as this is dependent on the student's approach.

2.4 The Approaches

In the previous section, it was noted that students' performance may be dependent on whether they engaged with the software boxes, that is, the extent to which the students attempted to make sense of the steps in the glass-box and open-box software. Of course, the students' performance may also be affected by whether they engaged in learning the mathematical topic and the tackling of the tasks. There is also a possibility that the software boxes may also affect the extent of their engagement with the tasks. For example, students may more readily use the black-box software, versus the open-box software, for calculations, as the black-box software is quick and easy to use. Thus, the way that students engage with learning the mathematical topic, solving the tasks and using the software boxes will affect their performance.

The way that students engage is considered to be the 'approach' in this thesis. Three approaches are identified but it is recognised that these approaches are not definitive of all the ways that a student may undertake when learning the mathematical topic, solving the tasks or using the software box.

The first approach identified is the 'processing level approach' (Section 2.5, p.27). The processing level approach characterises how students engage with learning the mathematical topic and the solving of tasks. The processing levels are defined as either being deep or surface level (Marton, 1975; Marton and Säljö, 1976). The 'explanation approach' (Section 2.6, p.32) is the second approach and it is typified by the quality and the quantity of explanations that students generate (Chi, Bassok, Lewis, Reimann and Glaser, 1989) in trying to understand the mathematical topic and solve the tasks. The final approach is the 'exploration approach' (Section 2.7, p.38) and it is exemplified by when and how often students decide to use the software boxes for solving tasks. It is useful to note that whilst these approaches are identified separately, they are not mutually exclusive to each other; for example, the processing level used by a student may influence the quality of explanations produced.

Finally, the approaches are only able to identify how or what a student is doing but this does not account for the students' affective and attitudinal factors towards mathematics in general which may also influence the extent of their engagement and hence their approaches. One affective factor is examined in this thesis, that of self-efficacy (Bandura, 1977; 1988) which is further discussed in Section 2.8 (p.42). Again, there are other influences that may affect students' performance on the software boxes, for example other affective factors such as mathematics self-concept or additional approaches such as on what part of the software box the student places more emphasis. However, self-efficacy and the identified approaches are assumed as the most pertinent for affecting performance but other affective factors and approaches will be drawn upon if needed.

The three approaches and self-efficacy are now discussed separately. Each section has an introduction to the topic and consideration is given to how each may influence performance, both generally and when students are using the software boxes. Where appropriate, discussions are made on how self-efficacy may affect the approaches undertaken by the students. The processing levels approach is discussed first followed by self-explanations and exploration approaches and then self-efficacy.

2.5 Processing Levels Approach

In this section, the extent to which students engage or process information when solving tasks is discussed and represents the first approach. Processing levels in this thesis are defined as either being surface or deep as identified by Marton and Säljö (1976).

2.5.1 Processing Levels: A Brief Background

Marton and Säljö determined these processing levels from a study involving 30 university students. In this study, each student was asked to read a passage on ‘curriculum reform’. Afterwards, the students were audio-recorded as they answered questions relevant to the passage. In particular, the students were asked to explain the passage (Marton, 1975). By analysing the transcripts, Marton and Säljö (1976) suggested that there were generally two types of processing levels, deep and surface. At a deep processing level, students based their understanding on the meaning of the learning materials, whilst in the surface processing level, students depended on memorising materials for reproduction (Richardson, 2005b). Marton (1975) further qualitatively assessed that the processing levels influenced the students’ performance: where a deep processing level was associated with a positive learning outcome, whilst a surface processing level led to a negative learning outcome. Laurillard (1979) contended that the use of deep or surface level processing was not an inherent quality of the student, but rather, changed depending on the learning context.

Based on the previous research, Ramsden and Entwistle (1981) extended the processing levels concept by suggesting that some students may have a preference for either the deep or surface processing level when reading for a course. They proposed that students may adopt a ‘deep approach’ or ‘surface approach’ when studying for a course. Using their terminology, research has developed into the area of ‘approaches to study’. The term ‘approaches to study’ is most often used when referring to the deep and surface processing levels in the literature. Using the ‘approaches to study’ term may prove confusing when referring to the terminology in this literature, that is, the three approaches. Therefore, to avoid confusion with the three approaches in this thesis and the ‘approaches to study’ research, the terms deep and surface processing levels or just processing levels will be used whenever referring to ‘the approaches to study’ literature where avoidable.

Through interviewing students on how they study, Ramsden and Entwistle developed a 64-item questionnaire which included items representing the processing levels. These items were statements which required students to use a Likert scale to indicate their extent of agreement. They named their questionnaire the ‘Approaches to Studying Inventory’ (ASI) which they administered to students at the end of a course. Biggs (1987) also extended the work of Marton and Säljö (1976) and developed a 42-item inventory which had items that measured the surface and deep processing levels of students reading a course. He called this the ‘Study Process Questionnaire’ (SPQ).

A score for each of the processing levels is found by summing its associated inventory items. Therefore, a student obtains scores for each processing level through these questionnaires. The processing level scores cannot be compared against each other since their final processing level scores only demonstrates whether a student tended to have a high surface or deep processing level.

2.5.2 Processing Levels: Mathematics, Tasks and Boxes

Both the ASI and SPQ measured processing levels generally for all courses. In some cases, these questionnaires have been modified to be subject specific. For example, Crawford, Gordon, Nicholas and Prosser (1998a; 1998b) modified the SPQ to measure processing levels in mathematics courses. This inventory was called the ‘Approach to Learning Mathematics Questionnaire’ (ALMQ). To test whether the processing levels impacted on performance in mathematics, Crawford *et al.* (1998a) administered the ALMQ to 127 students reading a first year undergraduate mathematics course. They found through a cluster analysis of the students’ performance scores (measured by their final examination mark) and the scores from the ALMQ that students with a high deep processing level score also had a high performance. This was similar to the results obtained by Marton (1975) who qualitatively assessed that there was a relationship between positive learning outcomes and the deep processing level for students studying the newspaper passage on curriculum reform. Therefore, Crawford *et al.* (1998a) were able to show empirically that processing levels were related to performance in mathematics. Hence, there is an expectation that students using a deep processing level should perform well on the three task types (i.e. mechanical, interpretive and constructive).

Crawford *et al.* (1998a; 1998b) found that students' conception of mathematics was related to their processing levels. Through analysing the open-ended responses on what students thought about mathematics, Crawford *et al.* developed the Conception of Mathematics Questionnaire (CMQ) to measure students' conceptions of mathematics. They considered that there were two types of conceptions, fragmented and cohesive. In the fragmented conception, students consider mathematics as numbers, rules and formulas which are applied to tasks. On the other hand, in the cohesive conception, mathematics is considered as a way of thinking for carrying out complex problem solving and for providing new insights into the understanding of the world. Students with a cohesive concept of mathematics may thus have a better conceptual understanding of mathematics.

Through factor analysis, the results from the CMQ and ALMQ in the study by Crawford *et al.* showed that there was a relationship between the two questionnaires' scales. That is, the surface processing level was linked to the fragmented conception of mathematics, whilst the deep processing level was linked to the cohesive conception of mathematics. Therefore students' processing levels provide an indication of students' mathematics conception and shed some insight on how the students will perform on tasks. In particular, the interpretive and constructive tasks will require students to have a more cohesive concept of mathematics to successfully solve them.

If students are asked to solve the three tasks, then it is expected that those with a deep processing level will perform better. However, how well students perform, when using the three software boxes for solving the three tasks, is uncertain. The reason for this uncertainty is that some students may have to process more information depending on which software box they are using and this may affect their performance.

For example, let us take a group of students who use mainly a deep processing level. If the students are assigned to either the glass-box or the open-box software, then they will use a deep processing level to understand both the mathematical steps and the task. However, students assigned to the black-box software only use a deep processing level for understanding the task, since there are no steps in the black-box software. Students assigned to the glass-box and the open-box software may become more cognitively or mentally fatigued than students using the black-box software. The tendency towards cognitive fatigue in this thesis is based on the definition by Trejo *et al.* (2007) and it is understood to be where students who are alert and motivated become more unwilling to continue undertaking mental work. Therefore, those who have cognitive fatigue are not likely to expend their cognitive effort in understanding the information presented to them.

The reason for this possible cognitive fatigue is that students using the glass-box and open-box software will have to find meaningful learning from the software steps shown, whereas the students using the black-box software would not have to do so. Therefore, this cognitive fatigue probably can affect performance using the glass-box and open-box software.

This analysis suggests that students using the black-box software will have an advantage and should perform better. However, there is also the possibility that using a deep processing level to understand the steps in the glass-box and open-box software could also help in the solving of the tasks. Or alternatively, the students may only use a surface processing level when looking at the steps in both of these software boxes and possibly then be able to solve these tasks with almost the same cognitive alertness as the students using the black-box software. Possibly, one way of knowing whether the students' cognitive fatigue is affecting performance is to investigate the explanations that students are generating for themselves and to ascertain whether these explanations incorporate cohesive mathematical concepts.

2.6 Self-Explanation Approach

This section now turns towards the approach of explanations, in particular self-explanations. The self-explanation process is where students generate explanations for themselves such as why things are occurring (Chi *et al.*, 1989). Through making these self-explanations, students are then able to gain understanding. Chi *et al.* observed self-explanations in a study where students were asked to think-aloud (Ericsson and Simon, 1984). They asked ten students to think-aloud as they studied worked-out examples and solved 19 tasks in Newtonian physics. However, they only analysed 3 of the tasks in qualitative detail because at that time they did not have the opportunity to systematically analyse the voluminous amount of data. Further, only eight of the ten students were included in the analysis. The first student was not analysed because of a poor self-explanation session and the other student was removed to have an even number of students analysed when they were split into groups. The 8 students were divided into 'good' and 'poor' students (top scoring 4 and bottom scoring 4 respectively). From the think-aloud transcripts, Chi *et al.* noted that the students were explaining the information for themselves as they solved the tasks. Chi *et al.* (1989) termed these 'spontaneous self-explanations'.

The think-aloud protocol was developed by Ericsson and Simon and is used to understand students' information processing such as the logical processes of task-solving. Smagorinsky (1998) put forward a major objection to the think-aloud protocol as he considered speech being uttered by a student as being culturally mediated. He asserted that speech cannot be looked upon as a window into students' minds since speech is ultimately a communicative or social tool and not a way to peek into the minds of people. Ericsson and Simon's (1998) rebuttal was that thinking-aloud had minimal reactive influences on students' thinking providing the students were asked to focus on the task and to just verbalise their thoughts. Ericsson and Simon pointed out that some students may think verbalising their thoughts is about explaining what their thoughts means. They thus suggested that students should do warm-up tasks to minimise this behaviour.

Chi *et al.* used the think-aloud protocol for identifying students' self-explanations rather than investigating students' information processing steps. The protocol provides a method for gauging students' sense making but could never be a true reflection of the sense making process (or any other protocol). For example, the verbalised thoughts may not be a true reflection as this could be a post-hoc rationalisation rather than an accurate representation of actual processes or that the sense making could be visual in nature rather than linguistic. Self-explanations may show how students interiorise (Piaget) or internalise (Vygotsky). In interiorisation, students construct their knowledge (Piaget, 1972; Tudge and Winterhoff, 1999) by creating meaningful or conceptual links between information gathered. Similarly, based on an experiment carried out with children, youths and adults involving verbal memorisation, Vygotsky (1931/1997) suggested that in internalisation, an adult learner uses an internal process through which the learner "actively makes connections between memorised words and the content … and organises the word into one pattern or another, etc." (p.186).

The verbalised self-explanation process is not an internal product, that is, it is made into an external product by speech. However, the verbalised self-explanations can give some insight into how the internalisation/ interiorisation processes might occur. Possibly, the verbalised self-explanations might change the internalisation/ interiorisation processes but the self-explanations do provide a way of understanding how the mind makes sense of information.

Self-explanations have also been determined via written self-explanations as used by Hausmann and Chi (2002) and Schworm and Renkl (2006), who found similar results to Chi *et al.* This method provides an alternative way of 'looking into the mind' but using pen-and-paper and as with speech, it is also an external product.

2.6.1 Self-Explanations and Processing Levels

Aleven and Koedinger (2002) suggested that self-explanation is a metacognitive activity that can help students' performance. According to Flavell (1976), metacognition refers to:

> one's knowledge concerning one's own cognitive processes or anything related to them, e.g., the learning-relevant properties of information or data. For example, I am engaging in metacognition if I notice that I am having more trouble learning A than B; if it strikes me that I should double check C before accepting it as fact. (p.232)

Therefore, through metacognitive activity, students become aware of their shortcomings whilst solving a task. Through the use of self-explanations, they are then able to make sense of the information. Chi *et al.* (1989) suggested that those students who made a high number of explanations (usually the 'good' students), that is, made more sense of the information, were able to perform better than those making a low number of self-explanations (usually the 'poor' students). Chi *et al.* also argued that the number of self-explanations was not a sufficient measure for performance but that good students (those students with high performance) generated high-quality self-explanations. High-quality self-explanations were where students were able to use knowledge from the subject domain to explain their answers. Chi *et al.* noted that those students who gave a high number of self-explanations were more likely to give higher quality of explanations.

Roy and Chi (2005) suggested that the quality of self-explanations could be divided into high or low explanations. With low-quality self-explanations, the students generally reread or paraphrase during the learning process whilst when students generate high-quality self-explanations, the students demonstrate the making of inferences, integrating statements and filling in the gaps when linking to the tacit knowledge (Roy and Chi, 2005). The criteria of high-quality self-explanations seemed to be related to a deep processing level and cohesive conception, particularly in having meaningful learning through making inferences.

This may imply that during learning, the students who are high self-explainers are the ones who are engaging in a deep processing level whilst those students who are low self-explainers are engaging in a surface processing level. Chi (2000) also referred to the high-quality self-explanations as being 'deep' explanations but she made no direct link to the deep and surface processing levels.

2.6.2 Self-Explanations and Tasks

Chi, De Leeuw, Chiu and Lavancher (1994) suggested that self-explanations, with respect to procedural tasks, facilitated students' construction of their learning by integrating their new information with existing information and hence allowed the students to improve on their procedural skills. This may reflect students using self-explanations to gain a relational understanding whereby they create new conceptual knowledge.

Further, Chi *et al.* (1989) noted that, when good students (students who did well on the tasks) explained, they tended to relate their explanations to the physics domain that they were studying whilst the poor students' explanations did not connect with physics principles or concepts. Similarly, if the students were studying mathematics then high performing students would most likely use mathematical explanations, whilst the students who were low performers would use other types of explanations. These other type of explanations may come from a non-mathematical domain and it may be based on knowledge that they have created from their life experiences. Therefore, the high performing students should be able to link their conceptual and procedural knowledge in a mathematical way, whereas the low performing students only used non-mathematical knowledge. As conceptual and procedural knowledge are linked, even though these tasks being used were mainly procedural (mechanical) tasks, this meant they were not precluded from integrating their conceptual knowledge.

Renkl (1997) also looked at spontaneous self-explanations provided for near-transfer and medium-transfer tasks in 36 students learning about probability. Near-transfer tasks were tasks that were similar to an example of a mechanical task provided and thus required students to use procedural knowledge. In medium-transfer tasks, the structure of the task was different and required students to generate a modified solution procedure (Atkinson, Renkl and Merrill, 2003). The medium-transfer tasks required students to recognise the different structure and make the link that the tasks could be solved in a similar way to the near-transfer tasks. This required some use of conceptual knowledge. Renkl (1997) found through multiple regressions that students' performance on medium-transfer tasks was better explained by the number of self-explanations than by their performance on near-transfer tasks. It seemed that more conceptual-oriented tasks required students to use more self-explanations, and thus perhaps for the constructive and interpretive tasks, more self-explanations will be expected for higher performance.

2.6.3 Self-Explanations and Software Boxes

Jones and Fleischman (2001) suggested that when students are presented with worked-out examples, they seldom try to understand them, because only when there is a gap in students' knowledge resulting from an impasse (a stumbling block) would they begin to spontaneously self-explain. Glass-box software provides a solution to a task that is quite similar to a worked-out example. Students may react similarly towards glass-box software and will not spontaneously self-explain when looking at the solution because they are not faced with an impasse.

However, although impasses allow students to spontaneously self-explain, Chi *et al.* (1994) suggested that there could be learning gains from prompting self-explanations, that is, a mechanism that asked students to self-explain. Most types of software are equipped with some kind of prompting mechanism which can tell students what they should do, such as through the help menu or a pop-up window. The prompting mechanism found in mathematical software does not necessarily prompt students to self-explain but perhaps could create an indirect prompt or an impasse which would encourage students to self-explain.

Black-box software might involve the least impasses as this software would solve the task without showing any steps. Glass-box software might act similarly to that of the black-box but perhaps, with a solution being presented to the students, there might be a higher incidence of students determining there was a gap in their knowledge. This gap is then the impasse and hence would prompt more self-explaining. Thus, when students are shown steps in the glass-box software, they might decide to self-explain what the steps mean and create a better understanding of the mathematical topic than those students using the black-box software.

The open-box software has the most impasses as the software mode requires the student to perform an action at each step. This impasse will allow the students to self-explain what is occurring at each step. However, the understanding gained by the students through the self-explanations of the steps in the software box may be mostly from procedural knowledge, although there may be also smaller gains in conceptual knowledge.

A similar argument to the one that was made about software boxes and processing-levels can probably be applied here. It is possible that in mechanical tasks students will self-explain more when using the glass-box and open-box software. However, this may make them more cognitively fatigued and hence unable to provide a high number or good quality self-explanations for interpretive and constructive tasks. The discussion above presumes that the students start with the mechanical tasks which would be most likely the course of action as this ensures that the students have familiarity in using the software box.

2.7 Exploration Approach

This section discusses the exploration approach which is used to examine how and when students use software for solving tasks under their own volition, that is, without a teacher intervention or not following a pre-existing procedure. By interviewing teachers, Ruthven, Hennessy and Brindley (2004) noted that software packages used by secondary school students in the subjects of mathematics and English were used for:

> supporting processes of checking, trialling and refinement, notably with respect to checking and correcting basic elements of work, testing and improving problem strategies and solutions, and editing and redrafting written texts. (p.271)

The key point to note from their interviews was that students used the software for testing and checking processes which implies that students were using software for exploring processes or scenarios. Similarly, Pierce and Stacey (2001) reported from a study of observing 30 students learning university calculus, that the students used CAS as an 'independent expert'. The students used the CAS to explore properties of functions for testing or making their own conjectures. For example, the students changed the values of coefficients and exponentials in mathematical functions to see the change in its corresponding graphs.

Trouche (2000) developed a classification of how students use technology (CAS or graphical calculators) when solving tasks. His category of 'overall calculator use' indicated how often students were using these tools under their own volition. Hence, his classification of students is dependent on how and when students explore with the CAS or graphical calculators. He provided evidence for his categories by observing a senior level high school class undertaking a design engineering project that covered calculus and elementary analysis. He classified students into five extreme categories: theoretical, rational, thinker, experimenter and scholar (see Table 4).

Table 4: Types of students based on their use of technology from Trouche (2000) (my translation)

Student	Information Source	Meta-cognitive activity	Privileged method of proof	Overall calculator use	Usefulness of calculator
Theorist	Notes	Interpretation	Analogy	Average	High
Rationalist	Pen/Paper	Inferences	Demonstration	Low	Low
Tinkerer	Calculator	Investigation	Accumulation	High	Low
Experimentalist	All	Comparison	Confrontation	Average	High
Scholastic	None	Investigation	Copy/ Paste	Average	High

Trouche indicated that when solving tasks, each type of student privileges specific information sources and calculator uses. For example, the theorists use references (notes, paper), work towards interpretation for understanding, use analogies for proof, spend about average time exploring on the calculator overall but their exploration time spent is usually fruitful. Trouche suggested that some students may have a predisposition as to how they used the software, that is, students had a particular style.

However, these particular styles of exploring with software by Trouche may be rather strategies that students employ depending on the topic. Coupland (2004) in her study investigated how students appropriated the use of Mathematica (a CAS). She issued both the ALMQ and a Mathematica Experience Questionnaire to her students. The former measured students' processing levels and the latter questionnaire measured the students' mathematical engagement with the software and their computing experience. Through examining the responses from 113 students, she noted that students' uses of the software were dependent on their processing levels. Her analysis showed that students with a deep processing level and a low computing background reported that they were still able to appropriate the tool to allow for mathematical engagement. In an earlier study, Laurillard (1979) found that processing levels may be dependent on the learning context. As Coupland showed that students' appropriation of software is influenced by their processing level, then it is possible that Trouche's student categories are not stable, that is, students may opt for any of these strategies depending on the subject or task.

Coupland also found that students' exploration with mathematical software using their own initiative was quite poor. Students were requested to mark on a visual analogue scale, a position on the line anchored by 'disagree' on the left and 'agree' on the right. The line was approximately 41 millimetres. With 113 students returning completed questionnaires, she noted for one item, "I often used Mathematica to explore my own questions about mathematics", that the students scored poorly. The students' mean score for this item was 10.3 out of 41 which indicated a high disagreement with this statement.

2.7.1 Exploration: Performance, Tasks and Processing Levels

Coupland's (2004) results suggest that the frequency of students' exploration with the software boxes may be low. A low frequency of exploration by the students might impact on students' performance on mechanical and constructive tasks. If students choose not to use the software boxes then they would more likely have procedural or arithmetic errors for both the mechanical and constructive tasks. Further, through exploration, students can test scenarios and via self-explanations, can build their conceptual understanding. This could potentially impact on their performance for not only the constructive tasks but perhaps also the interpretive tasks.

Moreover, Coupland found that students with a deep level of processing were more likely to choose to use the software for mathematical engagement than students with lower levels of processing. This might suggest that students with a deep level of processing would be more likely to explore with the software boxes for a purpose such as confirming answers or testing hypotheses.

2.7.2 Exploration and the Software Boxes

The frequency of exploration may also be dependent on the software box itself. Both glass-box and black-box software are able to solve procedural tasks easily as the student is only required to click the buttons to get the answer. However, the open-box software requires students to determine what they would do at each step and this may mean that students might be more reluctant to use this software for solving procedural tasks. This behaviour may thus impact on how students explore using the software boxes when solving the mechanical and constructive tasks. Students using the black-box and the glass-box software may then explore more for the mechanical and constructive tasks compared to students using the open-box software as there is a sense of more immediacy. As the mechanical tasks are relatively simple, that is, requiring only the inputting of values and executing a command, students may choose to always solve these tasks using the software boxes.

2.8 Self-Efficacy

The last aspect that is looked at which may influence performance is self-efficacy. Bandura (1986) defined perceived self-efficacy as “people’s judgement of their capabilities to organise and execute courses of actions required to attain designated types of performance” (p.391). The term ‘perceived self-efficacy’ is how students may view how well they can perform tasks. Whilst students who self-reflect through metacognitive activities about their ability to perform on tasks can increase their performance on tasks, Bandura suggested that this may also be a pit-fall for those students who produce “faulty thought patterns”(p.21). Depending on a student’s self-efficacy perception, they may consider the following when approaching a task:

- what to do
- how much effort to invest in activities
- how long to persevere in the face of disappointing results and
- how to tackle the task i.e. anxiously or self-assuredly (p.21).

Thus, students with a high self-efficacy when solving tasks will have more success since they will persevere until the task is solved. Often these students during the solving of the tasks will generate and test alternative forms of strategies (Bandura, 1986: p.391). Alternatively the self-doubters (those with low self-efficacy) will be more likely to abort their initial efforts if it proved to be deficient. As students with high self-efficacy have a higher tendency to test alternative strategies this suggests with respect to exploration, that students with high self-efficacy may explore more with the software by testing different numbers. However, this should be due to high perceived self-efficacy in the use of mathematics rather than in the use of computer software as Coupland (2004) found.

According to Schunk (1991), self-confidence is usually operationalised as the measurement of self-efficacy (see for example Gist and Mitchell, 1992; Pajares and Miller, 1994). Bandura (1986) suggested that high confidence students are students who are more likely to have high success and those with low confidence are more likely to be self-doubters. In fact, Collins (as cited in Bandura, 1986) carried out a study where students having high and low mathematical self-efficacy were given difficult problems to solve. He found that

> while mathematical ability contributed to performance, at each ability level, children who regarded themselves as efficacious were quicker to discard faulty strategies, solved more problems, chose to rework more of those they failed, did so more accurately, and displayed more positive attitudes towards mathematics. (Bandura (1986), p.391)

Pajares and Miller (1994) in their study of self-efficacy and mathematics performance using path analysis found that “students' judgments about their capability to solve math problems were more predictive of their ability to solve those problems” (p.200) than other variables they investigated such as gender, mathematics self-concept and students’ prior ability. Their study was conducted with 350 undergraduate students at an educational college where they were given a series of questionnaires which measured mathematics confidence, perceived usefulness of mathematics, mathematics anxiety, mathematics self-concept, prior experience and mathematics performance. Both the questionnaires for mathematics confidence and mathematics performance consisted of the same arithmetic, algebra and geometry mathematics tasks and two problem types (real and abstract). In the mathematics confidence questionnaire, students were asked to assess how confident they were in solving the task and in the mathematics performance questionnaire they solved the tasks. The mathematics performance questionnaire was administered to the students last.

2.8.1 Self-Efficacy and the Approaches

Thus far this chapter has suggested that processing levels, self-explanations and explorations affect performance. With the study by Pajares and Miller, students' academic self-efficacy or confidence is also seen as having a positive impact on performance. This suggests that there might be a relationship between the processing levels, academic self-confidence and self-explanations.

Duff (2004) conducted a study using the Revised Approaches to Study Inventory (RASI) that he administered to 244 business school students. The RASI is a revised version of the ASI but also includes items for measuring academic self-confidence. Through a correlation matrix, he was able to determine a relationship between high academic self-confidence and the deep processing level.

Whilst there are few studies using the RASI for mathematics, it is possible that students with higher mathematics confidence, similarly to the business students in Duff's study, would be more likely than students with lower mathematics confidence to engage in their work and try to make meaning out of their learning, that is, have a deep processing level. Students with low mathematics confidence on the other hand would be less engaged with the mathematical topic and thus adopt a surface processing level.

Therefore, if students with high mathematics confidence should have a deep processing level, this would probably mean that these students will then appropriate the software boxes for their mathematics engagement as was noted in Section 2.7.1 (p.40). Further, as high mathematics confidence students may probably be more au fait with mathematical concepts and terms, then there is a possibility that their self-explanations should be within the mathematical domain, whilst those of the low mathematics confidence students would be less in the mathematical domain.

2.8.2 Self-Efficacy and Attitudes to Technology

It might be worthwhile to investigate whether attitudes to technology might also influence performance.

In a bid to find out how technology influenced mathematics learning when these two are combined, Galbraith and Haines (2000b) developed a Mathematics-Computing Attitudinal Scale (MCAS) to find out. Their original questionnaire had six scales which looked at student's mathematics confidence, computer confidence, mathematics motivation, computer motivation, computer-mathematics interaction and mathematics engagement. Mathematics engagement was found to be highly correlated with mathematics motivation and the former was eventually dropped (Cretchley and Galbraith, 2002). A similar inventory was developed by Cretchley, Harman, Ellerton and Fogarty (2000) called the University of South Queensland Mathematics Technology (MathTech) questionnaire, which had three scales: mathematics confidence, computer confidence and attitudes to technology in the learning of mathematics.

Both of these inventories were administered to university students in differing technology programmes. Their results were quite similar (Cretchley and Galbraith, 2002), in that both of the questionnaires had low correlations between attitudes to mathematics and attitudes to computers. Further, Cretchley and Galbraith noted that from these two inventories the attitudes of learning mathematics with technology were more closely associated with the student's attitude towards technology rather than mathematics.

Pierce, Stacey and Barkatsas (2005) also sought to research attitudes towards mathematics and technology; however, unlike the previous two inventories mentioned, they applied their questionnaire in secondary schools and with a shorter number of items. Their questionnaire called the Mathematics and Technology Attitude Scales (MTAS) used items from both Galbraith and Haines and Cretchley *et al.* and measured on similar scales. In addition, the MTAS made use of items of another secondary school inventory which had items on mathematics and technology by Vale and Leder (2004) but whose focus was to find gender differences. The MTAS questionnaire had five scales which assessed student's mathematics confidence, their confidence with technology, their attitude of using technology for learning mathematics, affective engagement with mathematics and behavioural engagement with mathematics.

They found that students' attitudes to using technology for learning mathematics were dependent on gender and found for the males that this was positively related to their confidence with technology. However, for females, their attitudes to technology for learning mathematics were negatively related to their mathematics self-confidence. This phenomenon exhibited here by the secondary school students was not found in the studies at the tertiary level, where perhaps the gender attitudes towards mathematics or technology even out and are more equal across the genders.

Therefore, university students with a high technology background should not have an advantage in their performance over those students with a low technology background when it comes to using the software boxes and solving the tasks. However, the students with a high mathematics confidence will definitely have an advantage over those students with low mathematics confidence.

2.9 Analytical Framework

This chapter has explored how performance may vary depending on the task and the software boxes and the types of approaches that students may implement. Three approaches that students may undertake were identified: explorations, explanations and processing levels. The exploration approach was with respect to the software box use whilst the approaches to explanations and the levels of processing were from an individual task-solving viewpoint. The exploration approach was heavily dominated by whether students will use the software boxes and seemed only appropriate for the mechanical and constructive tasks.

Throughout this chapter, conjectures have been made to suggest there are connections between self-explanations, internalisation, conceptual knowledge, self-confidence and processing levels. These conjectures are presented in Figure 3 along with already established connections from the literature. As this figure shows, academic performance or learning outcomes (the centre of the triangle) is linked to the students' quantity and quality of self-explanations, academic self-confidence and their processing levels. The vertices of the triangle represent the processing level approach, explanation approach and self-efficacy as influencing performance.

It is uncertain where exploration may lie with respect to the other two approaches and self-confidence in the analytical framework. Exploration perhaps can influence performance but it is wholly dependent on whether the student chooses to use the software box appropriately. For now, the exploration approach is not included into the analytical framework and will be updated once sufficient empirical data are provided that suggests an evidenced connection. The additional concepts such as conceptual knowledge and internalisation are placed external to the triangle as these are considered to be cognitively linked. The concepts on the triangle vertices are measurable; however, the external concepts on the right hand side may not be easily measurable.

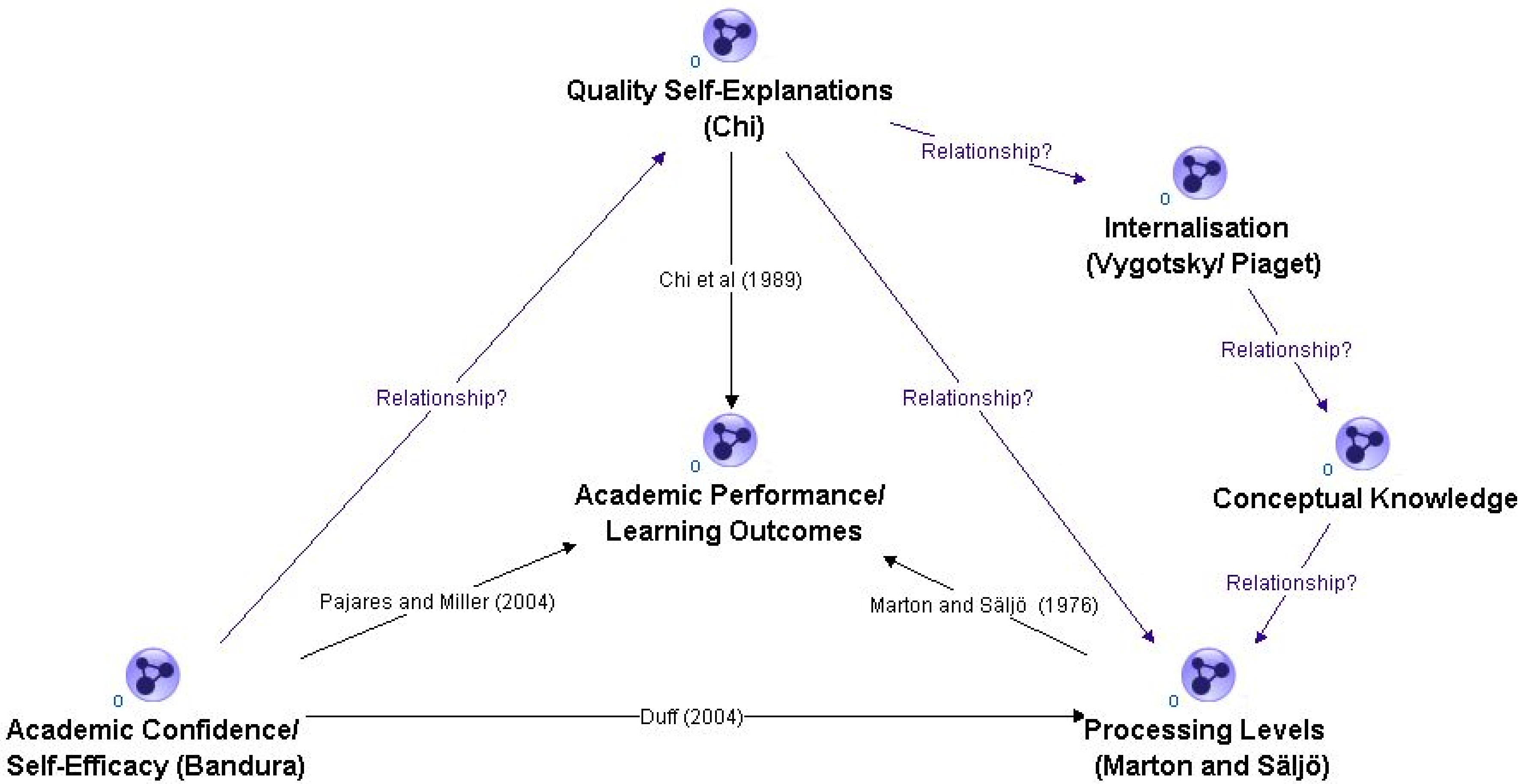


Figure 3: Analytical framework for performance and approaches in mathematics

The discussion so far has looked at how the understanding of mathematics might be evaluated from a cognitive view with respect to the software modes and the task types. As the task types were related to the conceptual knowledge, this would mean that the interpretive and constructive tasks which required conceptual knowledge would more likely need a deep processing level to be successful. As the processing levels were related to students' academic confidence, then the conceptual tasks may be influenced by the students' academic self-confidence. Thus, students' deep processing level on constructive or interpretive tasks might be dependent on their self-confidence (Figure 3). There is another dimension to the processing levels since they may also be influenced by quality of self-explanations being made by the students where high quality self-explanations should be linked to a deep processing level. The quality of self-explanations could perhaps be influenced by the level of confidence of the student.

Further, most of the studies quoted in Figure 3 were with respect to learning mathematics or mathematical-type of problem solving. Both Pajares and Miller (1994) and Crawford *et al.* (1998a) dealt with mathematics whilst Chi *et al.* (1989) used mechanical physics problems which is sometimes taught as applied mathematics. However, Duff (2004) study dealt with management education. Based on the research questions, this study would be mainly based on using the linkages on the sides of the triangle for analysing any data. Since the data being used in this study would be with respect to learning mathematics, it might be useful to determine if processing levels are related to academic self-confidence and thus by doing this, the study would confirm that the results found by Duff (2004) could be extended to mathematics as well.

2.10 Concluding Remarks

This chapter discussed two types of knowledge, conceptual and procedural, and how these are formed through understanding (Section 2.2, p.15). Further, the chapter indicated how students' performance can be measured on three task types which are representative of conceptual and procedural knowledge (Section 2.3, p.18). For each of the identified three approaches (exploration, explanations and processing levels), their influence on performance of the tasks and their effects on how students use the software boxes were explored (Sections 2.5, 2.6 and 2.7). Additionally, self-efficacy was also suggested as a factor in students' performance and how students used the software boxes (Section 2.8, p.42). Using the literature relating performance to the approaches, an analytical framework was created for analysing task-solving (Section 2.9, p.46). In light of this literature, the research questions are refined in the next section.

2.10.1 Another Look at the Research Questions

From Section 1.4 (p.8), the research questions are updated for the study investigated as follows:

How do students' approaches to the three task types (mechanical, interpretive and constructive) and their performance on these tasks depend on the software box they have access to?

1. *Does the students' performance in solving the interpretive and constructive tasks depend upon the software box the students have access to?*
2. *Do students' approaches to solving the three task types (mechanical, interpretive and constructive) depend upon the software box they have access to?*
3. *How are students' approaches to solving the three task types (mechanical, interpretive and constructive) associated with their performance? And does this association vary with the software box they have access to?*

Research Question 1 is reduced to two tasks rather than three since with the use of software, students should always compute the correct answer for the mechanical task. The approaches have now been identified as explorations, explanations and processing levels.

Explorations are drawn from the idea that students use the software boxes as an independent expert or for checking processes (Section 2.4, p.26) and hence exploration can be measured by judging whether students were using the software or not for these purposes. Further, students are able to self-explain and may draw on their knowledge domain when solving mathematical tasks either from mathematical principles or other principles such as real-life. The two levels of processes are the deep level and the surface level, and the students' use of one rather than the other can be measured through the ASI.

Chapter 3. Methodology

"At the beginning of all experimental work stands the choice of the appropriate technique of investigation."
- Walter Rudolf Hess

3.1 Introduction

The research questions set forth two aspects that needed to be measured or determined: namely, students' performance and their approaches when solving the three task types. The literature review indicated that performance was a measurement of individual cognition and that assessing scores on tasks can provide a performance indicator. However, measuring an approach was more challenging and the literature indicated that there were four aspects that had to be considered:

- How software boxes were being used (namely, the exploration)
- Students' level of cognitive processing demonstrated by self-explanations and processing levels
- Types of explanations such as real-life and mathematical explanations
- The self-efficacy of the student: that is, students' mathematics confidence

All of these four aspects were not considered independent of each other but rather mutually influencing the approach that a student took when solving a task. This chapter reports on the study that was undertaken to collect the data representing the scores and these four aspects.

This chapter begins with looking at the data variables related to the Main Study (Section 3.2, p.53). This is followed by a section on the summary of the design which discusses the Supporting Study 1, the two pilot studies and the Main Study (Section 3.3, p.57). The methods on how the data variables (including the approaches and self-confidence) are operationalised are outlined next (Section 3.4, p.66). A brief description of the remote observation method which was used for data collection is then provided together with the reasons for using this method (Section 3.5, p.78). Following this, a description of the sample and the recruitment of the participants are presented (Section 3.6, p.81). The chapter rounds off with a description of the mathematical domain used (Section 3.7, p.86), some brief notes on the transcription of the think-aloud data (Section 3.8, p.90), a discussion on the ethical considerations (Section 3.9, p.92) and some concluding remarks (Section 3.10, p.93).

3.2 Data Variables

Determining students' scores and approaches across the software boxes and tasks lent itself easily to an experimental design as students can be easily assigned to experimental groups. This design enabled the measurement of students' performance and observation of their approaches. A summary of the data variables that were collected for the Main Study, based on the literature is presented in Table 5. An additional variable, 'Problem' is included in this table. Problem is used as a way of organising the tasks and is further discussed in Sections 3.3.4 (p.62) and 3.3.5 (p.63).

The variables were grouped into independent variables, non-varying covariates, varying covariates and dependent variables. Independent variables were variables that the researcher manipulated such as the assignment of tasks and software boxes to students. Covariates were variables that this research had no control over but were present and may influence the study. The non-varying covariate was a variable that was not influenced by this research design, that is, the covariate remained the same (or near the same) throughout the study. Thus, a student's mathematics confidence was expected to stay the same for the duration of the study. The varying covariates, such as the propensity towards self-explanations, on the other hand were assumed to be dependent on this research design.

Table 5: Variables used for collecting data

Variables	Description	Data type
Independent Variables		
Boxes	Types of software	Categorical
Black-Box	Software does not show steps	
Glass-Box	Software shows steps	
Open-Box	Software allows interaction at steps	
Problems	Types of problems	Categorical
Problem 1	Toy manufacturing application problem	
Problem 2	Lumber manufacturing application problem	
Problem 3	Mathematically abstract problem	
Tasks	Types of Tasks	Categorical
Mechanical	Procedural knowledge using during solving	
Interpretive	Conceptual knowledge used during solving	
Constructive	Both procedural and conceptual knowledge used during solving	
Non-Varying Covariates		
Mathematics Confidence	Confidence of the student to do mathematics	Quantitative
Processing Levels (A)	The deep or surface processing levels that students take when solving all tasks	Quantitative
Varying Covariates		
Processing Levels (B)	The processing levels (deep/ surface)	Qualitative

Variables	Description	Data type
	that students use when solving each task	
Self-Explanations	The propensity of students to generate out-loud explanations when solving each task	Qualitative
Dependent Variable		
Performance	Scores that students have made on the interpretive and constructive tasks	Quantitative
Explorations	Frequency of using the software for testing numbers or conjectures for the mechanical, interpretive and constructive tasks	Categorical
Explanations	Types of Explanations	Categorical
Mathematical	Frequency of written mathematical explanations for interpretive and constructive tasks	
Real-Life	Frequency of written real-life explanations for solving the interpretive and constructive tasks	

The final variables were the dependent variables. The data collected for these variables were determined by an outcome of the intervention at different levels of the independent variables, which in this case were the solving of tasks when provided with a software box. The outcomes of the intervention were the scores on the tasks, categorising whether students explored with the software, and categorising self-explanations into real-life or mathematical explanations.

A mixed-methods methodology was employed (Creswell, 2003) for collecting data in this study. A mixed-methods approach is where both quantitative and qualitative data are collected for answering the research questions. Quantitative and qualitative data may either be collected concurrently or sequentially. Both the quantitative and qualitative data can then later be used for triangulation. In the concurrent method, both types of data are collected at the same time, whilst in the sequential method, either of the two data types are collected first then followed by the collection of the other data type.

In the experimental design employed for the Main Study, quantitative and qualitative data were collected concurrently. Through the use of the concurrent triangulation method, agreement between the quantitative and qualitative data was sought. In the experimental design, empirical data were collected for the tasks (performance scores, explorations and explanations frequencies, and processing levels) which aided in determining if there were any statistical differences or similarities amongst the software boxes. However, this data do not on its own shed any light on why differences or similarities might be occurring. More data were thus necessary to understand why the statistics found were significant and what influenced its significance. Therefore observations (audio/video recordings and note-taking) of what students were doing when solving the tasks were conducted. This qualitative audio/video and note-taking data were then triangulated with the statistical findings. This meant that even though statistical differences may show quantitatively that there were no differences between the software boxes, subtle variations in how students interacted with the software boxes were then able to be obtained via the observational data.

The use of quantitative data ensured that there was an extent of rigour and internal validity within the experimental design since statistical probability was used for determining quantitative variations (Hammersley and Atkinson, 1995; Howell *et al.*, 2005). According to Howell *et al.* (2005) validity is concerned with ensuring that what was being measured would be able to answer the research questions, that is, the instrument being used would accurately measure or represent the concepts. They further explained that internal validity referred to the rigour under which the study was conducted whilst external validity referred to the extent that the results could be transferable or generalisable to a population (see also Campbell and Stanley, 1963).

The study was based on university students and this meant that all variables could not be accounted for in an experimental design, since students' characteristics can not be held constant as their behaviour is constructed and reconstructed during the course of experiment (Hammersley and Atkinson, 1995). To account for this effect, the qualitative data were collected to gain insight and provide richer data on the students' behaviour (Hammersley and Atkinson, 1995; Savenye and Robinson, 1996), particularly when determining how certain approaches were used by a student in solving the tasks. With qualitative studies there is potentially more subjectivity than in the quantitative studies and thus precautions were made to minimise bias or at least be reflexive on how the interviewer/ observer influenced the data (Hammersley and Atkinson, 1995). Therefore in the observation notes, the researcher noted any activities that may have influenced the students (for example, notes were made when the researcher told the students of wrong data inputs into the software box). Further, in the transcripts, the researcher's comments and actions were also included.

3.3 Summary of the Design

This section gives an overview of the research design. Justifications and reasons why these design choices were chosen are given separately. In this research there were four studies, one supporting study, two pilot studies and one main study (Figure 4). The studies were named as Supporting Study 1, Pilot Study 2, Pilot Study 3 and Main Study in this thesis.

A protocol, similar to that of Renkl and Atkinson (2003), was used for Pilot Study 2, Pilot Study 3 and the Main Study. This section first gives an overview of Supporting Study 1 and then discusses the protocol used for Pilot Studies 2 and 3, and the Main Study. This is then followed by summaries of these remaining studies.

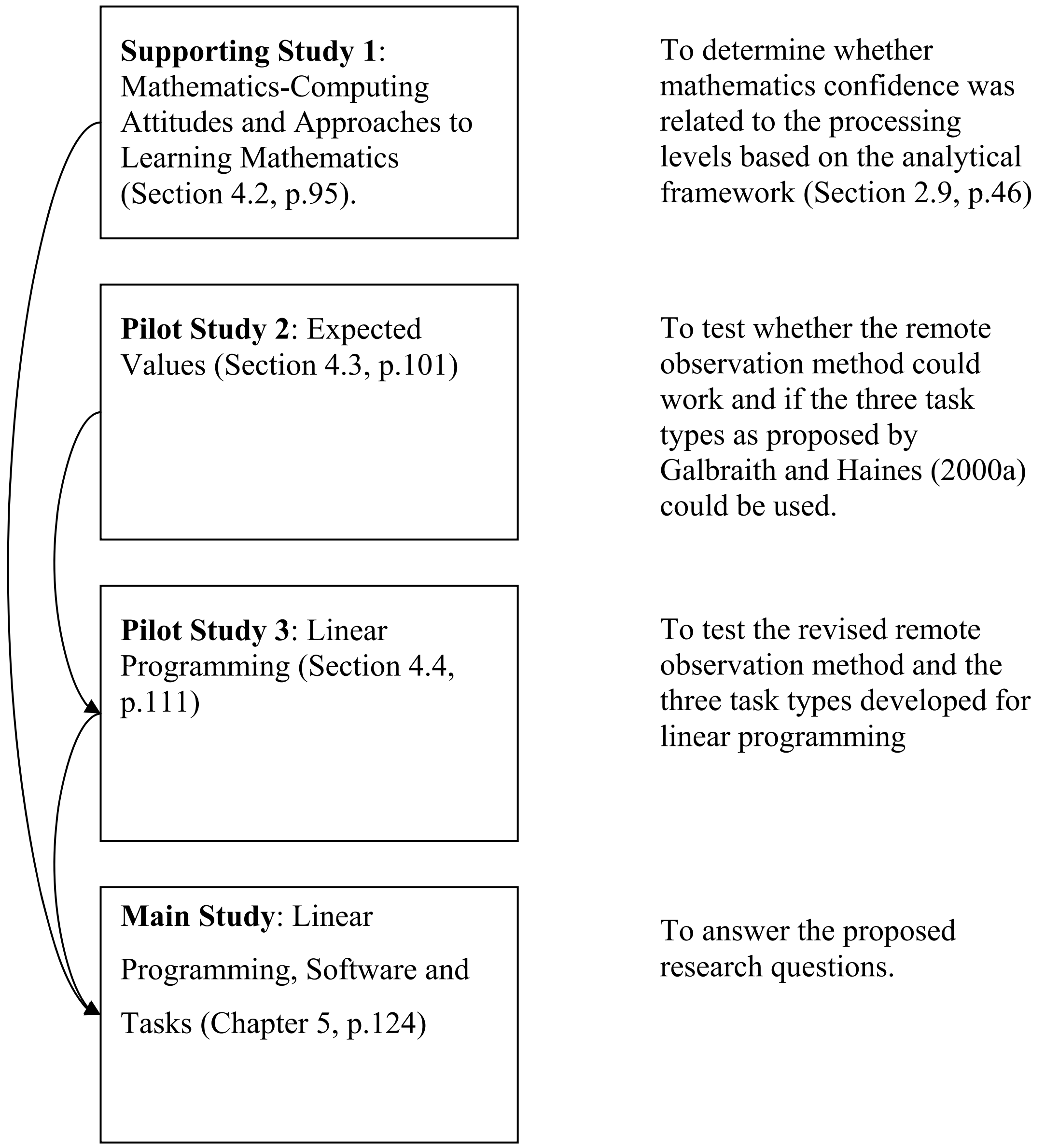


Figure 4: List of supporting, pilot and main studies and their purpose and connections

3.3.1 Supporting Study 1

The analytical framework, presented in Section 2.9 (p.46), showed, based on a study by Duff (2004), that there was a link between self-confidence and processing levels for management education students. No previous study was found that showed there was a link between mathematics confidence and processing levels. Supporting Study 1 was thus used to determine whether there was a link and to establish that the analytical framework in Chapter 2 was representative for mathematics students. The relationship was tested through a factor analysis of two inventories: the Approaches to Learning Mathematics Questionnaire (ALMQ) and the Mathematics-Computing Attitudinal Scores (MCAS). The methodology and the results for Supporting Study 1 are provided in Section 4.2 (p.95).

3.3.2 Protocol of Pilot Studies 2 and 3 and the Main Study

As there were several variables that needed to be measured, a protocol was established on knowing where and when each data type was collected. A scan through the literature of similar learning experiments with interventions when students were using software and doing mathematical problems, showed that a common method (e.g. Renkl and Atkinson, 2003) was a five step process. This process was adapted for the purposes of this study and Steps 6 and 7 were added (see Table 6).

The original process meant that students first filled in a background survey, then a pre-test questionnaire, followed by the learning materials, practice in doing some questions and the post-test questionnaire. Steps 3 to 6 were part of the experimental design stage (3.3.5, p.63). In some research (e.g. Große and Renkl, 2006), step 2 and step 3 were interchanged. This research was fundamental in finding out whether the use of software boxes yielded any difference in the understanding rather than the influence of instructional materials which was more common in other research (e.g. Renkl and Atkinson, 2003; Renkl, Atkinson and Große, 2004). The pre-test questionnaire was given in the third step. This meant that any prior mathematical knowledge before using the software boxes was ascertained and the influence from the instructional materials was minimised.

Table 6: Quasi-experimental protocol modified from Atkinson, Renkl and colleagues

Steps	Instructions
1.Background Questionnaire	Students are asked to fill in a demographic questionnaire, including questions asking for mathematical level, age and gender
2.Instructional/ Study Materials	Students peruse materials to understand the fundamental concepts required for the learning of the topic
3. Pre-test	Students took this test, from which the extent of their prior knowledge on the topic is determined before the influence of the intervention in the experiment. The pre-test tasks are at a lower difficulty level than the post-test tasks
4. Experiment	Students are provided with the interventions/ factors that are being studied
5. Post-test	Students work on a set of questions from which quantitative data are acquired and compared across the investigated interventions/ factors
6. ASI	Students were given one last questionnaire to determine their processing levels.
7. Debriefing	Students were allowed to ask questions related to this research, if they wished. This was an informal session.

Further, using the background questionnaire students' demographic factors were accounted for such as gender, age and level of mathematics attained. The affective factors acquired included students' mathematics confidence, computer confidence and MS Excel confidence. Excel confidence was measured as this was the software programmed to represent the three software boxes (Section 3.4.2, p.68). The students were asked to self-assess their confidence levels on a scale from 1 to 10 (where 1 = low and 10 = high). Whilst this self-assessment is not as reliable as using the mean of several self-assessed items relating to confidence, Bandura (1977) suggested that people were reliable in assessing their own self-efficacy.

During the experimental stage, the students were assigned to software boxes and a sequence of tasks. In Pilot Study 2, the students used all three software boxes; however, in Pilot Study 3 and the Main Study, students were only assigned to one software box (Section 4.4.1, p.113). Although the experimental stage, according to the table, suggests that the assignment of the intervention occurred after the pre-test was conducted, this was actually predetermined before the student arrived. However, it was only after the pre-test that the student saw their assigned software box. Their tasks, however, were attached to the end of the instructional materials but they were only asked to solve these during the post-test stage. Further, prior to answering the post-test, students were provided with a practice task to enable them to gain familiarity with the software box and practice the think-aloud protocol which was essential for collecting data on the self-explanations and determining their processing level (Section 3.4.4, p.74).

The details stated in the preceding paragraph were essentially what was done in the pilot studies (see Chapter 4, p.95), however, when it came to the Main Study an additional quantitative instrument was added to measure deep/surface processing levels.

3.3.3 Pilot Study 2: Expected Values

Pilot Study 2 (also called the Expected Values Pilot) was used for testing a new data collection method, testing the software modes and testing the three types of tasks. The data collection method was called remote observation and it involved collecting data on students' interaction with the different software modes (black-box, glass-box and open-box) via the internet. In this method, audio and video of the student is captured via microphones and web cameras, and their interaction on the software modes is captured through application sharing software. Application sharing software allows two or more remote users to view and use the same software application. For the remote observation method, the researcher used two computers for collecting the data.

Nine tasks on 'expected values' were created based on Galbraith and Haines (2000a) taxonomy (three of each type). The software modes were then developed and coded in MS Excel using Visual Basic for Applications (VBA) for solving the expected values tasks. Instructional materials were also created for understanding expected values. The students read the instructional materials; they then proceeded to learn how to use the software boxes and finally using the software boxes for solving the tasks.

In this pilot study, the software modes were tested to verify that they represented the software boxes and students were able to use the software boxes with minimum guidance from the researcher. Six students participated in Pilot Study 2 and tested all software modes and all nine tasks (one task type each per software box). Additionally, they also completed a pre-test on probability.

3.3.4 Pilot Study 3: Linear Programming

Based on the outcomes from Pilot Study 2, a new mathematical topic was chosen. This time, 'linear programming' was used because it was more complex to solve. Nine tasks were again recreated for this mathematical topic but now the three types of tasks were associated with a problem. That is, three problems were created each having a mechanical, interpretive and constructive task. These problems were referred to as Problems 1, 2 and 3. New software modes were coded in MS Excel using VBA for solving the linear programming problems. The software modes and the linear programming tasks were tested by three students.

Based on the outcomes from Pilot Study 2, each student was assigned to using only one software box. The reasons for this are twofold; firstly, using one software box reduced the cognitive effort required by the student in learning how to use all three software boxes. Secondly, using the one software box prevented students from forming a preference to a particular software box and thus becoming frustrated when having to use another software box, which they could see little point of using. The data were again collected via a remote observation method but this time with a new configuration of using only one computer. Both the new remote observation configuration and tasks were found to be acceptable and there were minor alterations to tasks where whole numbers were used instead of decimals.

3.3.5 Main Study

Using the experiences learnt from the Pilot Studies 2 and 3, the Main Study was designed as an experiment. The experimental design was a Latin Square design, where data on 38 students were collected. Approximately 12 students were randomly assigned to using each software box and solving the linear programming tasks. The same method as in Pilot Study 3 was used for collecting the data using similar tasks. For these students, the data were collected via the remote observation method. Thirty-six students were used from Trinidad and Tobago and two students from the United Kingdom (UK). Thus this meant that for any analysis for tasks, this required analysing 114 instances of the task type, that is, for example in interpretive tasks, there will be 38 students × 3 problems × 1 interpretive task = 114 interpretive tasks instances.

The experiment required assigning students to one of the software boxes and this was thus a 'between-subject' variable, referred to in this study as Boxes. 'Between-subject' refers to groups of students assigned to different conditions; in this case the condition is the software box.

Tabachnick and Fidell (2007) pointed out that students can become increasingly fatigued or practiced as they complete a number of similar tasks. For example, if students answered Problems 1 to 3, then students had sufficient practice in answering the preceding Problems 1 and 2, and may find Problem 3 simpler to answer. This means that students will have a higher score for Problem 3 than Problems 1 and 2. To counterbalance these effects, Tabachnick and Fidell (2007) suggested using a Latin-Square design. Campbell and Stanley (1963) indicated the advantage of the Latin-

Square was that it controlled for internal validity, although external validity may be lacking.

The problems were sequenced in three ways (see Table 7). Students were randomly assigned to one of the sequences. The variable representing the scores obtained from a particular problem sequence is referred to in this thesis as 'Sequence' and it is also a between-subject. In Table 7, 'Question 1' is the first problem solved by a student assigned to a particular sequence. For example in Sequence 2, the first problem solved by a student is Problem 2 and hence this is the student's Question 1.

The distribution of the students by software Boxes and Sequence is presented in Table 8. The modified protocol for conducted the Main Study is presented in Figure 5. The arrows represent the actions undertaken by or for the student.

Table 7: The three problem sequences

Sequence	Question 1	Question 2	Question 3
Sequence 1	Problem 3	Problem 1	Problem 2
Sequence 2	Problem 2	Problem 3	Problem 1
Sequence 3	Problem 1	Problem 2	Problem 3

Table 8: Distribution of students according to sequence and software box

Sequence	Black	Glass	Open	**Total**
Sequence 1	4	4	4	**12**
Sequence 2	5	5	4	**14**
Sequence 3	4	4	4	**12**
Total	**13**	**13**	**12**	**38**

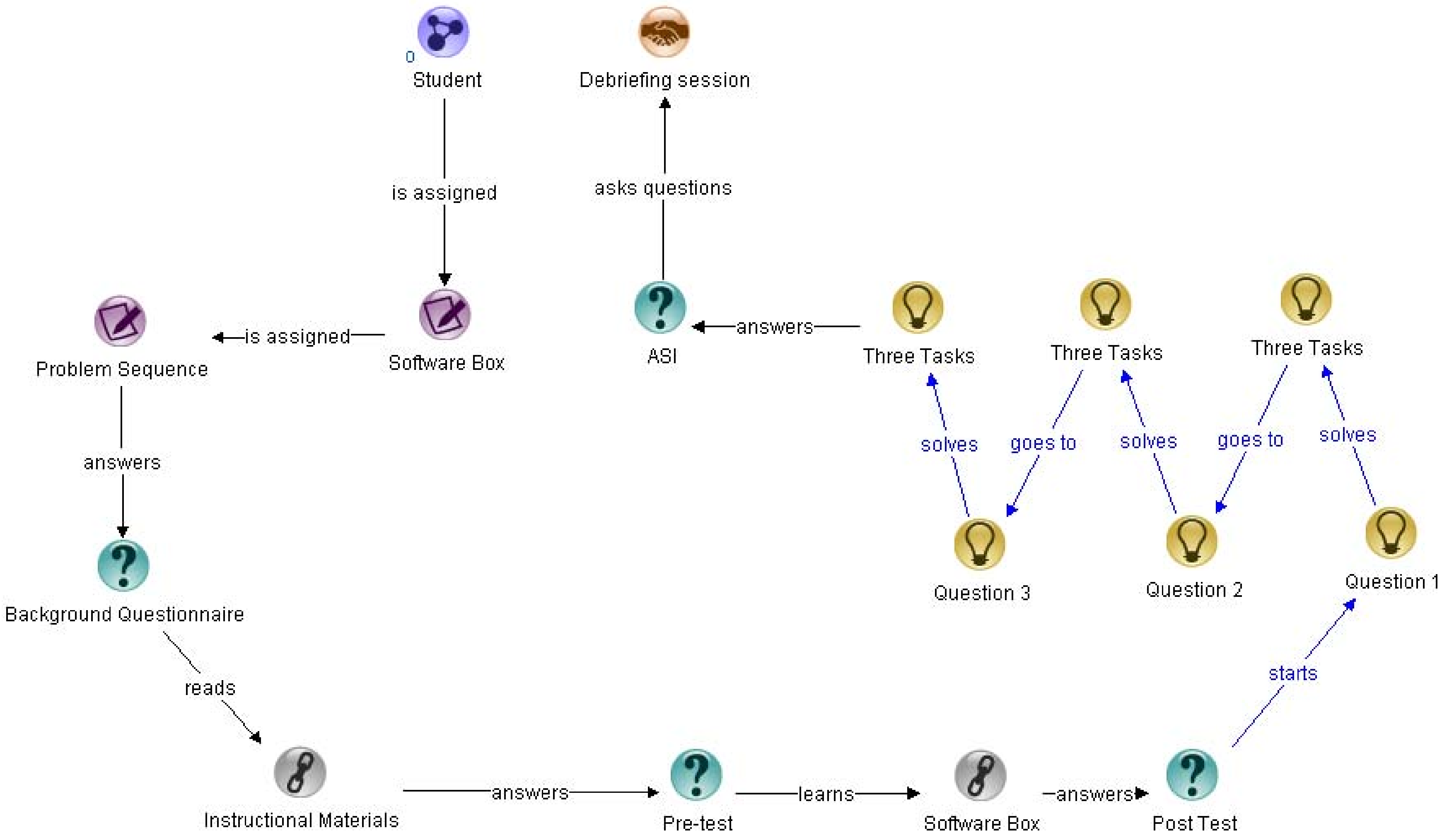


Figure 5: The protocol for conducting the Main Study

3.4 Operationalisation of the Variables

This section of the thesis looks at how the variables identified (namely, the software boxes and the approaches) were operationalised. Also considered in this section are the choice of mathematical domain and the development of the tasks.

3.4.1 Mathematical Domain

Whilst there were endeavours to gather tasks that measure conceptual and procedural knowledge from the literature, this proved futile and the Galbraith and Haines' (2000a) taxonomy was used for developing the tasks. However, this still meant finding tasks that were somewhat conceptual and procedural in nature. Tasks validated from previous research were sought as this would have provided a comparison between the data gathered in this study and the literature and also that there would have been less need to ensure that the tasks were able to measure conceptual and procedural knowledge.

Although the tasks from studies involving secondary schools were found to measure conceptual and procedural knowledge to some extent (e.g. Kadijevic and Krnjaic, 2003; Star and Seifert, 2006), these were inappropriate for the higher education context for this study. Other mathematical tasks used in higher education (e.g. Heid, 1988; O'Callaghan, 1998) were based on functions and calculus. Therefore in these topics, students were able to use both graphical and algebraic knowledge and translate from one to another. If the students are able to translate between these two types of representations from graphical to symbolic or algebraic representations, then this is considered a use of their conceptual knowledge. This meant that if these tasks (that is, from functions or calculus) were used, the influence of multiple representations had to be accounted for, which would have made the analysis difficult in determining whether the multiple representations (graphical, symbolic and algebraic) were the reason for the influence of conceptual knowledge or the mode of software used. Further, such topics as functions and calculus do not lend themselves easily to be processed in software as steps or for the tasks that do, they are quite trivial (see Figure 6).

$$\frac{d}{dx}\left(x^3+2x+1\right)$$
$$= 3x^2+2$$

Figure 6: Example of a trivial calculus question

As students with a wide range of disciplines were being used, this meant it had to be a task that students were not completely familiar with but yet arose naturally from secondary school mathematics. In the pilot study, the topic of expected values was chosen however these tasks were found to be trivial. As the expected value problem was trivialised students resorted to using pen/paper and making the need for the software boxes obsolete. Thus tasks needed to be developed from a domain where they can measure conceptual and procedural knowledge as well as lend themselves easily to being converted into the software boxes. Further, the steps could not be trivialised and as such algorithmic processes were looked into.

The algorithmic processes were needed that were sufficiently complex and thus whilst the algorithmic process may be purely mechanical which ensures only procedural knowledge, other questions relating to its applications and underlying mathematical methods may generate the conceptual knowledge required. As such linear programming's simplex method was chosen as it is used in various disciplines which meant that students from a wide variety of differing mathematical backgrounds would be able to it. Further, this was also a topic taught in the General Certificate of Education (GCE) Advanced Level (A-Level) examinations albeit in a special module and thus meant it was a topic that naturally arose from secondary school mathematics. Supposedly, other algorithms such as Gauss-Jordan elimination in linear algebra (on which linear programming is based) could also have been used but it did not afford the same advantage of being popular in various disciplines.

3.4.2 Problems and Tasks

Whilst one of the concerns was ensuring that the tasks were not trivial and that they were related to the three task types, the only challenge left was deciding on whether the linear programming problems should be application-type or mathematically abstract-type problems. Hennessy (1999) has cautioned against the use of real-life contextual tasks as they do not provide students with a better grasp of mathematical concepts, and students may be confused where there may be several meanings and understandings. However, linear programming mathematical models were developed based on a real-life context during the 1940s and it was used as an efficient way to solve complex planning problems during World War II (Encyclopædia Britannica, 2009). Generally real-life or application type problems are taught first to the students as a way of understanding linear programming concepts in both courses (for example, all three of the Open University courses containing linear programming, MU120, BM240 and M373, are taught in this manner) and in textbooks (see Winston, 1994).

Linear programming was thus introduced to the students through the instructional materials in a real-life context. The issue raised by Hennessy is still valid and thus the real-life application linear programming problems in the instructional materials and the post-test all dealt with receiving profit through production (agricultural, toy and furniture production) in order to keep the contexts similar. One problem was chosen as being abstract. As these were university students, the use of application or real-life contexts would have been familiar to them. Further, except for the linear programming problem introduced in the instructional materials, students were first asked to solve the linear programming model without any context and then the context was provided afterwards by indicating what product the variables represented.

Through providing the problem before providing context, it also ensured that students did not wrongly formulate the linear programming model.

Problems from Winston (1994) were chosen and modified to fit with the mechanical, interpretive and constructive tasks. Each interpretive and constructive task had two parts, one part where students provided a solution or answer and the second part where they gave a detailed explanation for their solution/ answer.

Whilst there were several aspects of linear programming that could have been chosen, only the simplex method, that is, the algorithm used for solving linear programming problems was chosen (see Section 4.4.1, p.113). Two application problems were chosen and one abstract problem. All three tasks were assigned to each problem and the reason for this was to reduce the monotony of doing the same thing as this was one of the complaints in Pilot 2: Expected Values pilot (Section 4.3.3, p.109). The chosen problems are listed in Table 9.

Table 9: The three problems and their associated tasks

<u>Problem 1 (Toy Application Problem)</u>

a) Mechanical: Solve

$$\text{Max } z = 2x + y$$

s.t.

$$2x + y \le 100 \quad \text{(constraint A)}$$

$$x + y \le 80 \text{ (constraint B)}$$

$$x \le 40 \text{(constraint C)}$$

b) Interpretive: Now, x refers to the no. of toy trains manufactured and y refers to the no. of toy soldiers manufactured whilst constraint A refers to painting hours and constraint B to carpentry hours. Interpret what this solution means to the toy company who wants to maximize their profit by producing toy trains and toy soldiers. Provide as detailed an answer as possible.

c) Constructive: If the profit per train has increased by £1, how would this affect the number of toy trains and toy soldiers being sold and why? Provide as detailed an answer as possible.

<u>Problem 2 (Furniture Application Problem)</u>

a) Mechanical: Solve

$$\text{Max } z = 30x + 15y + 10t$$

s.t.

$$8x + 6y + 2t \le 48 \text{ (Constraint A)}$$

$$8x + 4y + 3t \le 40 \text{(Constraint B)}$$

$$4x + 3y + t \le 16 \text{ (Constraint C)}$$

$$y \le 5 \text{(Constraint D)}$$

b) Interpretive: Let x = no. of desks manufactured, y = number of chairs manufactured and t = number of stools produced. Let also Constraint A = number of hours available for carpentry/day (i.e. building the product), Constraint B = feet of lumber available, Constraint C = the number of hours/day available for finishing (i.e. painting and polishing the product) and D is the demand for the number of chairs. Which product(s) was not produced and give the possible reason(s) why? Give as detailed an answer as possible.

c) Constructive: If the number of hours available for carpentry/day is increased from 48 to 60 hours, how would this change what the Furniture Company manufactured and why? Give as detailed an answer as possible.

Problem 3 (Mathematical Abstract Problem)

a) Mechanical: Solve

$$\text{Max } z = 6x + 8y + 13t - u$$

s.t.

$$\begin{aligned} 3x + 4y + 6t - u &\leq 0 \text{ (Constraint A)} \\ 2x + 2y + 5t &\leq 100 \text{(Constraint B)} \\ u &\leq 90 \text{(Constraint C)} \end{aligned}$$

b) Interpretive: Why do we allow linear programming to have ≤ constraints rather than just < constraints? Which variable will we not want to have a high value for? Give as detailed an answer(s) as possible.

c) Constructive: If u can be made greater than 90, what is the largest value that it can be? And why that value? Give as detailed an answer as possible.

3.4.3 Software Boxes

There was no known mathematical software that was able to represent or carry-out all the actions of the three software boxes. Although there were separate software packages that represented each of these three boxes, a mixture of software were not considered because:

- There was no guarantee that these software types would solve the same type of mathematical domain tasks, that is, finding for example three software boxes for solving geometry or algebra
- If there were three software types representing the boxes that solved tasks within one mathematical domain, then this raised the issue whether the students' performance and approaches determined from the boxes were reliable and valid since the students' performance or approach can be attributable to the user-design rather than ability of the software to show and interact with steps or not.

As such, MS Excel was the software used, and it was programmed through Visual Basic Applications (VBA) to represent the characteristics of the black-box, glass-box and the open-box software. MS Excel was chosen as it was familiar to many students and thus minimized the effect that familiarity with other types of software might have on the learning of the topic. Further, using the VBA, answer sheets were developed in MS Excel to allow students to type their answers for the tasks. This reduced the risk of transcribing and inputting answer data incorrectly during the analysis stage.

For the linear programming software boxes, the decision on which steps to include in the black-box, glass-box and open-box software (see Section 4.4, p.112) and the interface design had to be decided. The interface was dependent on the number of Problems given, as each Problem was programmed on an individual Excel sheet. Also, five buttons were created for each Excel sheet: 'Input Problem', 'Iteration', 'Reset', 'Answer Form' and 'Clear All' (see Figure 7).

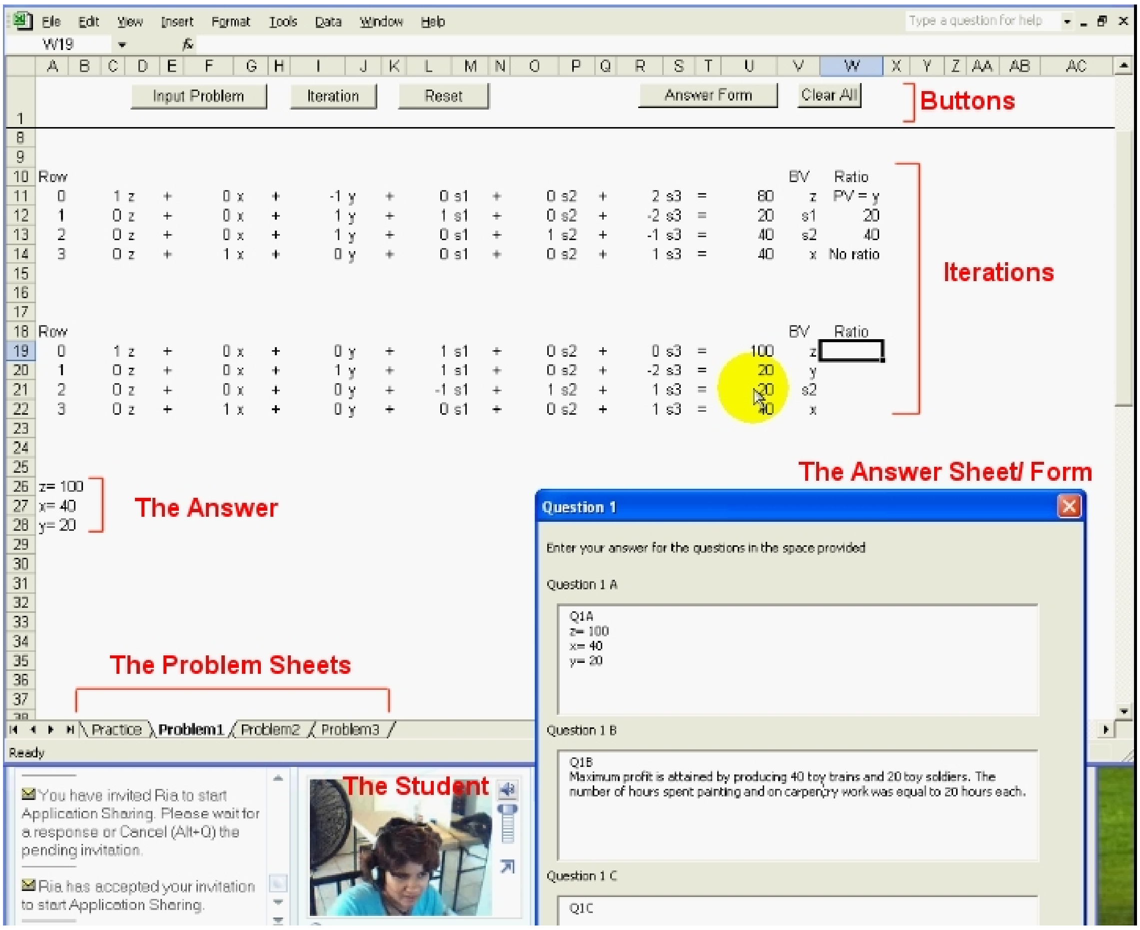


Figure 7: The glass-box software for solving linear programming which was developed in Excel

The ‘Input Problem’ button allowed the students to input values. To solve the problem, the students clicked the ‘Iteration’ button. If it was a black-box the answer came up immediately without showing the iterations. However, for the glass-box, every time the student clicked Iteration, an iteration was shown. The student had to click iteration until a pop-up came up saying a solution was found. This pop-up was necessary to ensure that the student knew the VBA linear programming procedure had ended. The ‘Reset’ button was used for students to clear all the iterations but not their inputted values, whilst the ‘Clear All’ button erased the iterations and inputted values. The students used the ‘Answer Form’ button to enter their answers. There was one Answer Form per sheet.

3.4.4 Explanations

Whilst the student's approach to using the software was captured via the web cameras and screen recording in the remote-observation method and the scores from the intervention in the experimental design, thus far, little has been said on how the self-explanations of students were captured during the solving of the tasks. Recall from Section 2.6 (p.32) that self-explanations are defined as explanations that students generate for themselves whilst learning or solving tasks. The think-aloud protocol developed by Ericsson and Simon (1984) was thus used where students were asked to verbalise their thinking processes out loud. From this, the spontaneous explanations that students generate for themselves were captured and analysed. Whilst written explanations could have been used, because of the remote observation method employed in collecting data (see Section 3.5.2, p.79), this could not be used for technical reasons, although there was a possibility that students could have typewritten them. However, instead when it came to students answering the problems based on the Galbraith and Haines (2000a), students were asked to give detailed answers and this in some way also represented the written explanations.

Further, there was concern as to whether it was necessary to record students' actions beyond what they were doing on the computer and what they were saying, for example recording whether they were writing on paper and reading the instructional materials. However, based on the pilot study, recording of the students' actions were not necessary during the analysis if the students were able to think-aloud and say exactly what they were doing. For students who were more reticent, the recording of their actions allowed the researcher to know where their focus was. The recording of these actions were taken in two ways: the researcher made written observation comments with a time-stamp and also through the video recording of the students. The former was considered a less reliable method for recording the actions but when suitably reconstructed with the screen-capture videos provided further insight through triangulation.

3.4.5 Processing Levels

In addition to the think-aloud protocol from which levels of processing could be determined, a 10-item Approaches to Study Inventory (ASI) used on the Social and Organisational Mediation of University Learning (SOMUL) project (Edmunds and Richardson, 2009; Richardson and Edmunds, 2007) was given to the students after they solved all tasks. This instrument is referred to as the SOMUL ASI in this thesis and it is used for quantitatively measuring whether students were using a deep or surface processing level more predominantly during the session (Section 2.5, p.27).

The 10-items on the SOMUL ASI was chosen by Edmunds and Richardson (2009) from an unpublished 'Approaches to Learning and Studying' scale developed at the University of Edinburgh. Several inventories were considered for measuring the deep/surface processing levels. A shorter inventory than the traditional 64-items ASI by Ramsden and Entwistle (1981) was investigated since 64-items would be time-consuming when added onto the experimental session which took between 1½ and two hours.

Therefore inventories and questionnaires with a low number of items were considered including, an 18-item ASI questionnaire (Gibbs, Habeshaw and Habeshaw, 1988), 12-item SOMUL ASI questionnaire (Richardson and Edmunds, 2007), 20-item Study Process Questionnaire (SPQ) by Biggs, Kember and Leung (2001) and the 26-item Approaches to Learning Mathematics Questionnaire (ALMQ) (Crawford *et al.*, 1998a; 1998b). The shorter questionnaires were more satisfactory as they were less time-consuming and hence the 20-item SPQ and 26-item ALMQ were eliminated. The 18-item questionnaire was not considered either as its internal consistency was previously questioned by Richardson (2000). Although, there were concerns on whether the 12-item SOMUL ASI questionnaire would be sufficient for determining whether there was deep or surface processing levels, Richardson and Edmunds (2007) showed that the items loaded satisfactorily on the factors of deep and surface processing levels, except for two items each relating to either a surface or deep learning. These two items were dropped and a 10-item SOMUL ASI was used for this research instead.

3.4.6 Explorations

Whenever students used the software boxes for testing numbers or decided to redo a procedure to examine a process, this was termed 'exploration' (Section 2.7, p.38). Only for the mechanical tasks were students required to use a software box. The constructive tasks were devised to be solved by either pen-and-paper or a software box. The interpretive task was devised to be answered solely by pen-and-paper.

By examining the students' videos and observing what they were doing for each task, students were either coded as not-exploring (0) or exploring (1) for each task. Each video was checked at 5-10 second intervals to determine how the students were using the software. For mechanical tasks, students were coded as exploring if they used the software boxes for any other purpose besides solving the given mechanical task such as solving using a different number. In the case of the open-box software when students tested a different sequence of processes whilst solving the tasks such as using a different pivot variable, this was also coded as exploring. If the students used the software boxes for either the interpretive or the constructive task this was coded also as exploring.

3.4.7 Mathematics Confidence

As noted in Section 2.8 (p.42), mathematics self-confidence is used as an attitudinal measure in this thesis. Bandura (1986) pointed out that self-confidence should be measured as close as possible to the time period when performance is measured. He also stated that self-confidence should be measured before performance as students' evaluation of their prior experiences influences how they will perform in future tasks.

There are several instruments for measuring mathematics confidence including the Mathematics-Computing Attitude Scales (MCAS) by Galbraith and Haines (2000b) or the Mathematics Confidence Scale (MCS) by Fennema and Sherman (1976). These instruments have a high level of internal consistency but use at least 30 items. However, during the 2-hour session, the experimental intervention required the students to learn linear programming, understand a software box and then solve nine tasks in linear programming using the software box. A scale was thus needed that will ensure the least fatigue for measuring mathematics confidence before the students proceeded to the experimental intervention.

Bandura (1986) has suggested that students are quite capable of judging their own levels of confidence accurately but he cautioned that the assessment of confidence should be tied closely to the mathematical topic rather than a global assessment as in this case general mathematics. The linear programming topic was chosen because of its unfamiliarity to students. Whilst algebra was considered to be the closest topic to linear programming, some tasks required the use of logic and understanding word problems. Therefore, a general assessment of mathematics confidence was used to encompass all these areas.

Students were asked to assessed their level of mathematics confidence on a scale of 1 to 10, where 1 = low confidence and 10 = high confidence. Collins in Bandura (1986) also used a similar method where he asked students to assess themselves on whether they had high and low mathematics confidence and he found through using this method that students' performance was related to their assessed confidence levels. As this simplified scale was used, students were also asked to assess their computer confidence and their MS Excel confidence using a similar scale.

3.4.8 Performance

Performance was measured by the marks acquired by students for each task. A marking scheme was developed for all problems and tasks and is presented in Appendix 6 (p.313). For the mechanical tasks, all students were required to solve these via the software boxes. As the researcher ensured that all students inputted the correct values for the mechanical task, this meant all students got these tasks correct. It was imperative that students solved the mechanical tasks correctly to ensure they stood a chance of solving the interpretive and constructive tasks since both the interpretive and constructive tasks were linked to the mechanical task solution. Therefore, scores from mechanical tasks were not included when investigated performance differences between the software boxes.

3.5 Remote Observation

The remote observation method was used for collecting data in this study. This section deals with why remote observation was chosen and a brief overview of the remote observation process. This process is also further discussed in Chapter 4.

3.5.1 Choice of Remote Observation method

Given that this study was based at The Open University, which is a distance and online university, finding students to participate at the university was a challenge. As such remote data collection methods were investigated in which data can be collected from students in various geographic locations. Therefore, a method was needed where geographic location would not hamper data location particularly in collection time and set-up. Hence, web-conferencing remote observation was developed and used. In web-conferencing remote observation, participants interacting with software were observed via the internet by employing web cameras for voice/video conversation and application-sharing facilities which were usually bundled into web-conferencing software.

Alternatively, data-logging remote observation (see Holzinger, 2005) could have been used. In the remote data-logging observation, special software is loaded onto the participants' computers whereby keyboard and mouse clicks may be recorded and later collected for analysis. However, data-logging does not provide rich video data and further, any data collected using this method has to be returned to the researcher either through electronic or postal means. The web-conferencing remote observation process lent itself to both a qualitative and quantitative data collection. Firstly, this method allowed the collection of both video and voice data and provided a richer analysis than data-logging remote observation. Secondly, higher numbers of students could be compared if an on-site and user-lab observation was used. In the user-lab situation, the participants are invited to a lab where the participants interact with the software and can be observed by the researcher. The on-site observation is similar except that the researcher goes to the participants and observes them whilst they work on their computers.

3.5.2 Details of Remote Observation method

Application-sharing facilities allowed the researcher to share the software, in this case, the VBA-programmed MS Excel worksheets across the internet, and allowed the students to have control in interacting with the software. Either Windows Live Messenger (with Windows Messenger application sharing facility) or Skype with Unyte application sharing were used as the web-conferencing software. Students were thus required to have either Windows Live Messenger or Skype installed in their computer in order to participate in the study. Two set-ups of the web-conferencing remote observation were investigated, one using a two-computer configuration and the other a one-computer configuration. These are further discussed in Pilot Study 2 and Pilot Study 3 respectively.

Moreover, the experimental design and the protocol of the study were easily enabled through the web-conferencing remote observation method. As this method, used the internet, the background questionnaire, the pre-test and the SOMUL ASI were created and delivered via webpages to the student. Participants were then able to enter the answers online and these were then submitted directly to the researcher. This helped in getting an electronic copy of the data which reduced the need for transcribing or inputting data. Moreover, unlike the data-logging remote observation method, there was no dependence on participants' conscientiousness in emailing the data. However, any additional workings that participants did such as scribble or sketch on pieces of paper were lost unless saved and sent to the researcher. Although technically participants could use sketching software which they can application-share as well, this would not have been natural as paper to them.

Further, the remote observation was good for allowing the researcher to observe participants in their naturalistic environment and therefore allowing the participant the comfort of using their own equipment without creating any anxiety in operating new equipment. However, this was only the case for the data collected for the UK students in the Pilot Studies and the Main Study. Students were also recruited from Trinidad and Tobago for the Main Study through gatekeepers. Gatekeepers are used in this context to refer to persons who helped the researcher (Hammersley and Atkinson, 1995), by recruiting students in their university. The gatekeepers also set up a remote observation laboratory site in which students were brought to instead. Whilst initially it was interesting to investigate the students in their natural environments, the remote observation method allowed the setting-up of remote laboratories and observing people in varied geographic positions in the world.

Also, as the participants were in their own environment or perhaps in a different environment to the researchers, there was no overwhelming issue of power relations balance (Hammersley and Atkinson, 1995), that is, the research environment was not completely controlled by the researcher as it would have been in an user-lab situation. Further, this remote observation method allowed with minimal extra cost to extend the population or sample group to participants in different parts of the country or in different parts of the world.

Using this method, problems involving logistics were minimized as the researcher's personal computer was employed and all that was needed was an arrangement for a virtual meeting time with the participant. Further, the researcher effect was also minimised as the researcher to some extent had some freedom in being able to react to facial expressions to what the participants were saying without adversely affecting what the participant was doing, as participants were not likely to see them through the web camera once they started working on the application-sharing software. Further, the researcher was able to make notes without making the participant anxious about what was being written about them, as they were unlikely to see the researcher.

3.6 Sample

For the pilot studies, the participants that were used were from a purposive sample and intended to test the instructional materials, the tasks, the software and the remote observation method. The participants chosen for the pilot studies were all postgraduate students doing either their masters or PhDs. For Pilot Study 2, the 6 students who participated were all from the Open University. These students were approached personally by the researcher and asked whether they will participate in Pilot Study 2. After these students showed an initial interest, an email invitation was sent to them. For Pilot Study 3, three students were also approached personally and were asked to participate in the study. This time only one student was from the Open University and the other two students were from elsewhere. There were more strictures set for the selection of participants in the Main Study which is further explained in the next section.

In the Main Study, based on the experimental design, the expected number of students was 36, although data from 38 students were collected (see Section 5.2.1, p.125). Whilst it might be argued that this yielded only a small number of participants, the sample size of 38 students was comparable to those used in other experiments on mathematical problem-solving at campus-based universities. For example, Renkl and Atkinson (2003) reported sample sizes of 34, 54, 45 and 28 in experiments that they conducted over several years.

Further, using a software developed by Lenth (2006), it was determined that 12 students for each software box (that is, 36 students in all) should provide a statistical power of more than 0.8. Power indicates the ability of a statistical test to find a difference, if there is actually a difference and it is dependent on the number of groups and the sample size (Howell, 2002). Power values of over 0.8 are generally considered to be good.

3.6.1 Accessing Students via the Internet

Initially, these students were expected to be UK undergraduate students as the research main concern was investigating university students' learning mathematics. Thus, a wide sample of UK undergraduate students was intended to be recruited as this would ensure that the results would be more generalisable. The UK students were recruited through social networking websites and students forums such as Facebook and the Student Room.

It was expected that recruiting students from their popular social networking sites such as Facebook would be easy. However, this was not a straightforward or viable method of recruiting students. After writing to 14 UK Facebook university communities to post a recruitment notice on their forums, only 9 gave permission, and no students were recruited from these. Subsequently, paid advertisements targeting university students were used on Facebook. These advertisements appeared as electronic flyers on the 'Home Page' of each Facebook user. The flyers targeted participants based on their social network (e.g. university), age and gender.

Three paid flyers were used which had a web link and offered a £10 Amazon.co.uk voucher. Each flyer was for 1000 views by targeted Facebook university students. These students were chosen by reviewing 184 UK Facebook university communities. From this 104 UK Facebook university communities were selected based on having: a) at least 300 members and b) universities with mathematical disciplines. For each advertisement, 50 universities were chosen at random to proportionally represent Scotland, Wales and England (the Northern Ireland university communities' numbers were low). Using this process, there were only three responses from potential participants of which only one participated. Additional posting on other internet student forums did not help either. Two gatekeepers at the University of Reading and the University of London posted advertisements on their electronic notice boards, but this yielded no contact from any students.

There were perhaps several factors that influenced why students were not coming through these avenues; firstly it was probably due to the computer equipment requirements that were asked for such as web-cameras and microphones. Secondly, as the study was hoping to use some type of advanced mathematics, it was necessary that students had completed General Certificate of Secondary Education (GCSE) Mathematics or equivalent and some kind of mathematical type subject at a higher level either at A-level or at the university level for example in economics, statistics, chemistry or physics. Students from all disciplines were recruited as the study wanted to have a wide selection of students as possible, so it can be more generalisable to the student population.

Dwyer, Hiltz and Passerini (2007) also recruited participants through Facebook forums for their survey (they did not indicate how many forums); they were able to recruit 69 Facebook participants in exchange for a music download. Thus, it is possible for surveys to recruit participants via Facebook. Possibly asking for students to take part in a study requiring them to provide a web camera and microphone reduced the chances of finding students. Clough (2005) in her online survey of mobile learning had a high response rate when she posted a link on an online mobile technology forum. It seems plausible that if the survey is on a topic that the participants are enthusiastic about, then the participants are more likely to respond to internet advertisements. Given that the topic being advertised was mathematics, this maybe was not the best subject to find enthusiasts to take part in the study. Even so, one participant (Participant 33) took part because he was quite interested in how the Excel files were coded in VBA. The researcher sent him a copy of the Excel file after his participation in the study.

3.6.2 Recruiting Students via Gatekeepers

However, to counteract these problems and accessing these students became an issue. The next step was to find gatekeepers (Hammersley and Atkinson, 1995) that could aid in finding these students through snow-balling techniques (Sapsford, 1999), that is, by making contact through persons who knew undergraduate students such as colleagues and friends. As the methods (via internet and snow-balling techniques) for recruiting UK students failed considerably, a large portion of the students were recruited from the University of the West Indies in Trinidad and Tobago. One gatekeeper was able to access a large group of students. The choice of using remote observation as a way of collecting data also made this possible without any large technical difficulties.

The use of gatekeepers at the University of the West Indies proved to be successful. The past working relationship between the researcher and the gatekeepers was instrumental in the finding and recommending of students. These students were recruited personally rather than through advertisements. Further as a mini-laboratory (for the remote observation) was set up by the gatekeeper with the required equipment of a microphone and web camera, this made it easier to find students to take part in the study as they did not have to provide their own equipment.

The gatekeepers only had a small pool of students whom they could personally ask and who were willing to do mathematical tasks. For example, after the second gatekeeper exhausted all her own students, the second gatekeeper found a third gatekeeper to help provide the last few students (9) for the study. These were the students who were mainly from engineering, mathematics and physics.

Further, the students recruited via the gatekeepers were remunerated with chocolate bars at the end of the study, although they were not told in advance that they would receive chocolate bars. This showed that remuneration was less important in finding participants than having good gatekeepers who could personally ask students to take part and find students who were comfortable with the subject area.

Additionally, as the students in Trinidad and Tobago had a similar secondary school and tertiary system to the UK, this meant that results might be appropriate for UK students as well. In Trinidad and Tobago as in the UK, there is a secondary school examination that students take at age 16, which is called the Caribbean Examination Council (CXC) examinations which are equivalent to the GCSE. Further, at age 18 students also need to matriculate through the UK General Certificate of Education (GCE) at the A-level or a local A-level called the Caribbean Advanced Proficiency Examination (CAPE, administered also by CXC) but the majority of the students take the GCE A-level. Further, the requirement for the university level is also similar to the

UK. According to Association of Commonwealth Universities (2008), in Trinidad and Tobago the university requires students to possess 2 GCE A-level subjects and 3 further subjects at the CXC level. This is similar for the UK except where GCSE passes in the subject levels are required. At the university level, the degree award system is the same since the University of the West Indies was initially an external college of the University of London until 1962 when it was granted charter to award its own degrees.

3.7 About Linear Programming

As linear programming (LP) is the mathematical topic that would be used in understanding mathematical software, it is necessary to understand what it is and studies that have been conducted using software in the teaching and learning of LP. First of all, LP is a mathematical principle that seeks to find the best possible solution for a set of linear inequalities. It is often used to maximize profit or minimize cost in the production of items which is subjected to a number of constraints. Figure 8 illustrates a LP problem for a toy store producing soldiers and trains with labour hour constraints and it is similar to Problem 1 in the post-test.

There are usually three main topics in LP, the formulation of the problem, the solution to the problem and the sensitivity analysis. In the formulation of the problem, the students are required to take a word problem and transform this into a list of linear inequalities (Figure 8). In the solution to the problem, students are required to solve this problem either by hand or software. By hand these methods are the graphical method or the simplex algorithm. The graphical method, which is used for two variable problems, requires the students to draw the linear inequalities and find the optimal solution by using the intersection of these lines, a process similar to the graphical solution of simultaneous equations (see Figure 9).

Linear Programming Problem:

Giapetto's Woodcarving Inc. manufactures two types of wooden toys: soldiers and trains. A soldier sells for \$27 and uses \$10 worth of raw materials. Each soldier that is manufactured increases Giapetto's variable labour and overhead costs by \$14. A train sells for \$21 and uses \$9 worth of raw material. Each train built increases Giapetto's variable labour and overhead costs by \$10. The manufacture of wooden soldiers and trains requires two types of skilled labour: carpentry and finishing. A soldier requires 2 hours of finishing labour and 1 hour of carpentry labour. A train requires 1 hour of finishing and 1 hour of carpentry labour. Each week, Giapetto can obtain all the needed raw material but only 100 finishing hours and 80 carpentry hours. Demand for trains is unlimited but at most 40 soldiers are bought each week. Formulate a mathematical method of Giapetto's situation that can be used to maximize Giapetteo's weekly profit.

Solution:

S = no. of soldiers sold /wk
T = no. of trains sold /wk

Objective Variables

$$\text{Max } Z = 3S + 2T$$

Objective Function

s.t.

$$\begin{aligned}
2S + T &\le 100 \quad \text{(Finishing)} \\
S + T &\le 80 \quad \text{(Carpentry)} \\
S &\le 40 \quad \text{(Demand)} \\
S, T &\ge 0
\end{aligned}$$

Constraints

Figure 8: An example of a formulated linear programming problem

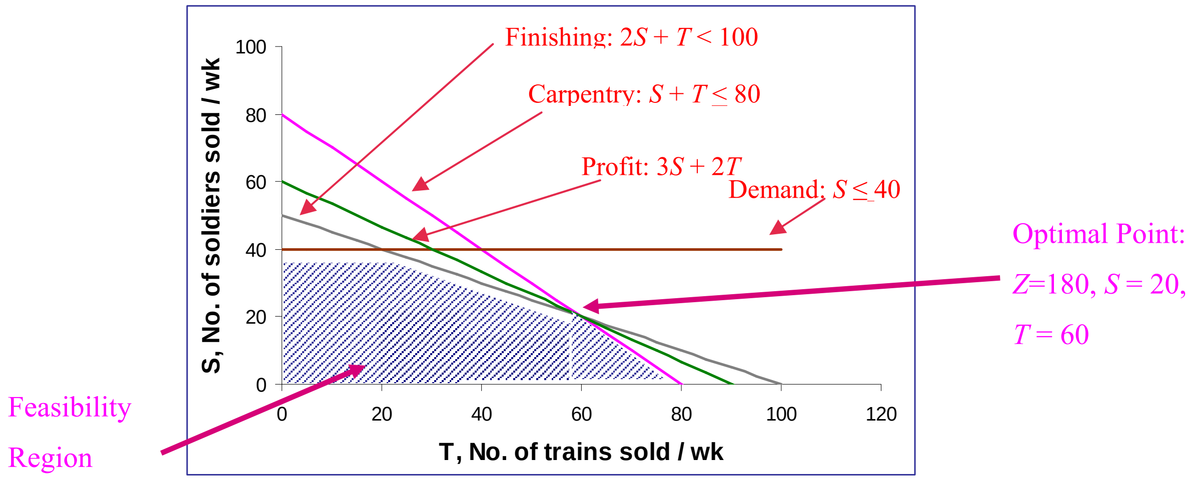


Figure 9: Illustration of the graphical method for solving linear programming problems

The simplex algorithm is an iterative process where students use similar principles to the linear algebra elementary row operations to solve the problem. In the simplex algorithm, several iterations have to be performed. First, however, all the inequalities are made into equations by adding something call a slack variable (the slack variables are represented as *S1*, *S2* and *S3*). The equations are then manipulated by placing all variables on the left hand side (LHS) of the equal sign and all numbers are placed to the right hand side (RHS) of the equation. This generates what is termed the canonical form which is the first shaded portion in Table 10.

Table 10: Simplex algorithm method for solving linear programming problems

Row Calculations	*Z*	*S*	*T*	*S1*	*S2*	*S3*	RHS	BV	Ratio	Row
	1	-3	-2	0	0	0	0	*Z*	PV = *S*	0
	0	2	1	1	0	0	100	*S1*	50	1
	0	**1**	1	0	1	0	80	*S2*	80	2
	0	1	0	0	0	1	40	*S3*	40	3
3*R3 + R0	1	0	-2	0	0	3	120	*Z*	PV = *T*	0
-2*R3 + R1	0	0	1	1	0	-2	20	*S1*	20	1
-R3 + R2	0	0	1	0	**1**	-1	40	*S2*	40	2
1*R3	0	1	0	0	0	1	40	S	N/A	3
2*R1 + R0	1	0	0	2	0	-1	160	*Z*	PV = *S3*	0
1*R2	0	0	1	1	0	-2	20	*T*	N/A	1
-1*R1 + R2	0	0	0	-1	1	1	20	*S2*	20	2
	0	1	0	0	0	**1**	40	*S*	40	3
	1	0	0	1	1	0	**180**	***Z***		0
	0	0	1	-1	2	0	**60**	***T***		1
	0	0	0	-1	1	1	**20**	***S3***		2
	0	1	0	1	-1	0	**20**	***S***		3

The simplex algorithm then involves students choosing a variable that would increase the value of the profit (*Z*). The chosen variable is called the pivot variable (PV) and the value of this variable is increased through the iteration. For the first iteration, this variable is usually the one with the highest positive coefficient in the objective function or the most negative coefficient when the objective function is changed to an equation (in this case *S*). Using this variable, elementary row operations are performed by manipulating the equation rows (R) either by the multiplication or addition of row coefficients from which a new set of equations are formed. The effect of increasing the pivot variable changes the coefficients of all the equations and the profit of *Z*.

The iteration occurs again with these new set of equations, from which the pivot variable is chosen from the new set of equations. Each iteration in the table is shaded. The term ‘basic variable’ is also used in linear programming and was mentioned to the students. This term refers to the variables for which a value was found for it. Thus, in the example provided when the last iteration was performed, the basic variables would have been *Z*, *T*, *S3* and *S*. More information on linear programming is available in Appendix 5 (p.304).

There have been few studies that investigate the teaching and learning of LP. Some studies such as that by Albritton, McMullen and Gardiner (2003) and Hosein (2005) have used surveys of lecturers to investigate how LP is taught but few have looked empirically at how students learn LP except for Smith (1994). In her investigation of students learning LP with a CAS software called DERIVE, Smith found that students thought studying lecture notes and handouts were the most helpful in understanding LP. The students however found that working with computer individually and computer demonstrations were not very useful.

It is unclear whether the approach of teaching could have influenced the learning preference of the students and their views on the use of computers. Perhaps also, the way the computer was integrated into the teaching of LP into the class may have influenced students’ opinion of software in the learning of LP. The preference of students for handouts and lecture notes may indicate an information transmission approach by the students (Kember, 1997). Therefore, it may be useful to see how the studying approach may influence students' use of software. Further, since the course targeted business and social science students, probably their attitudes towards CAS may have been more negative, as spreadsheets were preferred in these disciplines (Hosein, 2005). Thus, students probably may have different attitudes towards different software depending on their discipline and this probably could influence their preference of studying approach.

3.8 Transcription of Think-Aloud Data

For the analysis of the results obtained in the Main Study, the students' answers were coded (see Section 5.4, p.136) and also scored. Further, eight students' video/ audio data were transcribed whilst students' speech and actions (using paper or working on the software) were transcribed; the researcher's remarks were transcribed as reported speech. Predominantly the researcher prompted the student to keep talking and had minimum conversational exchange with them during the solving of the tasks. The transcripts were used for determining students' self-explanations or actions and not their interactions with the researcher. Hence, it was not considered necessary to include the students' minimal interaction with the researcher within the transcriptions except as researcher's summarized actions such as "after being prompted".

The marking scheme is presented in Appendix 6 (p.313). Students' answers were coded for explanations within MS Excel by indicating whether the explanations were mathematical or real-life. The answers were then imported into NVivo and were coded according to problem, task, software, and student characteristics such as mathematics confidence and gender. Coding in this way allowed easy access through a query to answers that various group of students were making and hence a wide range of evidence could have been provided when discussing issues presented in Chapter 6 (p.174).

Students' answers however were corrected for spelling mistakes, for example changing 'soliders' to 'soldiers' and correcting text speech such as 'd' to 'the' when this data are presented in Chapter 6. The original answers are included in the attached CD.

The transcripts of the 8 students were colour coded to allow easy analysis, since description of the video was also provided. The colour codes employed were black for actual speech, red was used for speech that the researcher was uncertain about due to internet interruptions or mumbling on part by the student, blue for recording what the students did or what the researcher said and brown for the typed in answer, for example:

> *1:05:10: Turns back to the instructional materials, "Chairs ... [mumbles] ... give as detailed an answer as possible ... ok. Why wasn't it produced? The z is whaat?", looks at the screen briefly, and begins to resume typing his answer, "reasons being 1)" and then fixes it to read, "reasons being maybe 1)", continues to type his answer.*
>
> (Participant 30, M, GB, Higher MC = 8)

This allowed easier scanning of the speech made by students when thinking-aloud and determining their procedure in answering problems such as checking the instructional materials or looking at the screen. The participant codes provided with this student quote is explained in Table 11.

Table 11: Participant Code Key

Code	Representation	Alternative Codes
Participant 30	Participant Identification Number	
M	Gender	M = Male; F = Female
GB	Software Box	BB = Black-Box; GB = Glass-Box; OB = Open-Box
Higher MC = 8	Level of Mathematics Confidence and the associated score	Lower MC = Low Mathematics Confidence; Higher MC = High Mathematics Confidence; "= 8" is the mathematics confidence score

3.9 Ethical Considerations

For Supporting Study 1, approval for conducting the online survey on The Open University (OU) students was granted by the OU's Student Research Project Panel (SRPP). The SRPP reviewed the questionnaires and also asked the researcher to receive permission from the course managers.

In the Pilot and Main studies, students were all asked to append their names to electronic consent forms which detailed information about the study and what was expected from them. Appendix 5 presents the consent form used in the Main Study which was similar for the Pilot Studies. In the Pilot Studies and the Main Study, the researcher answered all questions of the students pertaining to study and research with openness.

For the Pilot and Main studies, students were asked permission to allow their images or videos be presented in conference or written material (for example in this thesis). Only one student in the Main Study expressed concerned about the possibility of her image being used elsewhere but was comfortable with being recorded for data analysis purposes only. This student was assured that her data will only be viewed by the researcher.

Upon consultation with researchers in the educational technology field, ethical approval from the OU's Human Participants and Materials Ethics Committee (HPMEC) was not sought for any of the studies. The reason was that the main ethical consideration of HPMEC was whether 'any reasonable judgement would suggest that no harm could arise to any person, living or dead' and this was the case in this research.

In Supporting Study 1, students were required to fill in educational surveys which were completely anonymised and this data were held securely and confidentially. Therefore, there was no reasonable suspicion that this will harm the students.

With regards to the Pilot Studies and the Main Study, the only reasonable harm that was adjudged was perhaps normal fatigue arising from using the computer. Therefore, length of time at the computer was kept at a minimum with the expected participation period lasting for 1½ to 2 hours. This was not an unreasonable time limit as computer-based examinations such as the Scholastic Aptitude Test (SAT) and Graduate Record Examination (GRE) usually last about three hours and are more cognitively demanding than this current study.

All video, audio and observation data from the students were held securely on a server and on an external hard-drive. Whilst transcriptions of eight students from the Main Study are provided in this thesis (see attached CD), the identities of the students have been concealed by using Participant numbers. Any of the students' images used in this thesis is for software use illustration and there is unlikely to be any reasonable harm arising from the use of these images. Note again that the students gave consent for their images to be used in academic work.

3.10 Concluding Remarks

The chapter outlined briefly Supporting Study 1, the two pilot studies (Pilot Study 2 and Pilot Study 3) and provided detailed information on the choices and experimental design made for the Main Study (Section 3.3, p.57). A Latin-Square experimental design was also employed for the Main Study. This meant through statistical analysis of the quantitative data collected that the variances amongst the software boxes can be computed.

The chapter also showed how the Main Study variables were operationalised to collect both quantitative and qualitative data for performance and the three approaches: exploration, explanations and processing levels for the tasks and software boxes (Section 3.2, p.53). Exploration is measured by how much students used the software boxes, explanations by whether the students used real-life or mathematical explanations and processing levels via the ASI.

The web-conferencing remote observation method was discussed which demonstrated why it was developed and how it helped in the collection of the students' audio/video data via the Internet whilst students solved the tasks on the software boxes (Section 3.5, p.78). Issues regarding the recruitment of participants via social networking sites and the use of gatekeepers were highlighted (Section 3.6, p.81) along with ethical considerations in particular with respect to the remote observation method (Section 3.9, p.92).

Chapter 4. Supporting and Pilot Studies

"I would say in just about every investigation we have, there will be differences of opinion, where you have partial facts, as to what those facts mean."
-Robert Mueller

4.1 Introduction

In this chapter, one supporting study and two pilot studies are discussed and are labelled as studies 1, 2, and 3. The Supporting Study 1 was used to determine whether mathematics confidence was related to the deep processing level as indicated in the analytical framework (Section 4.2, p.95). Pilot Study 2 was used mainly to test the remote observation process set-up and to test the three software boxes programmed in Excel (Section 4.3, p.101). Pilot Study 2 used expected values as its mathematical topic and is referred also as the Expected Value study. Pilot Study 3 (Linear Programming Pilot) was used to test the chosen linear programming tasks to ensure that the participants were able to solve them (Section 4.4, p.112). The chapter presents the study design and the results from the three studies and also addresses the implications of the pilot studies for the design of the Main Study (Section 4.5, p.121).

4.2 Supporting Study 1: Mathematics Confidence and Processing Levels

Supporting Study 1 was intended to verify one aspect of the analytical framework developed in Section 2.9 (p.46). From the literature, Duff (2004) showed that academic confidence was related to the deep processing level, however this was in management education not in mathematics. This study thus aimed to investigate whether this relationship held true for mathematics.

Duff used the Revised Approaches to Study Inventory (RASI) to measure deep and surface processing levels for management education students but the Approaches to

Learning Mathematics Questionnaire (ALMQ), which had been constructed for mathematics students, was better suited for this current study (Section 2.8, p.42). The ALMQ did not have a scale for measuring mathematics confidence but the Mathematics Computing Attitude Scales (MCAS) did (Section 2.8.2, p.45). The advantage of the MCAS was that it included scales for computer confidence as well as mathematics-computing confidence and it would be of interest to determine whether the deep/surface processing levels that a student undertake might influence this as well, particularly as the current thesis was interesting in understanding how students were problem-solving with software. Thus, using both inventories meant that factor analysis could be used to determine if the deep processing level and mathematics confidence were loading on the same factor and hence from this infer that they would be related to each other.

One aspect of this study was to ascertain students' confidence and attitudes towards mathematics when using software and thus students were needed from a selection of disciplines using a variety of mathematical software. Thus, students were chosen from various courses such as psychology, mathematics, statistics and technology. All the students were currently enrolled on distance-education courses at the Open University. Table 12 shows the distribution of students. The 1800 students were selected by the Survey Office at the Open University based on their guidelines. Their guidelines ensure that the survey was not during a student's examination period as well as making certain that the students were not selected recently for another survey.

This population provided students with a reasonably wide variation in mathematics confidence. The students were chosen based on the software used in their courses and were equally distributed across the three categories of software (600 each). The three categories of software were computer algebra systems (CAS) which were typified by MathCad, spreadsheets (usually Excel) and statistical software which included SPSS, Genstat and Minitab. All of these software types were black-box in nature as black-box is the most popular type.

Table 12: Number of students surveyed from the 11 courses

Course	Subject	CAS	Spreadsheets	Statistical	Total
D841	Psychology	0	0	28	28
DD303	Psychology	0	0	250	250
M248	Statistics	0	0	193	193
M346	Statistics	0	0	129	129
MS221	Mathematics	203	0	0	203
MST121	Mathematics	230	0	0	230
MST209	Mathematics	122	0	0	122
T207	Technology	45	0	0	45
T305	Technology	0	292	0	292
T308	Technology	0	98	0	98
U316	Technology	0	210	0	210
Total		600	600	600	1800

4.2.1 Online Questionnaire

An online questionnaire consisting of two inventories was sent to the 1800 students. The first questionnaire was the ALMQ (Crawford *et al.*, 1998a; 1998b) which was based on the Study Process Questionnaire (SPQ). This questionnaire measured the deep and surface processing level scores of the students and used a 5-point Likert scale from 1 for ‘only rarely’ to 5 for ‘almost always’. The second questionnaire employed was the MCAS (Galbraith and Haines, 2000b). The original MCAS used a seven point linear scale ranked from 1 for ‘strongly agree’ to 6 for ‘strongly disagree’, which required students to mark their answers along a visual analogue scale. The inventory was modified for use in this Supporting Study to be a 5-point Likert scale from ‘strongly agree’ to ‘strongly disagree’ as it needed to be coded for a webpage. Using this scale meant that lower scores in the 6 scales were related to students having higher mathematics confidence, mathematics motivation, computer confidence, computer motivation, computer-mathematics interaction and mathematics engagement. Whilst for the ALMQ it was the opposite, higher scores meant higher deep processing level and surface processing level scores.

In the MCAS, there were items that were negatively scored and hence the polarity was reversed when the scale was calculated. Many of the items in the MCAS referred to the use of computers; this was replaced with the term ‘software’. Most Open University students used computers to access their online conferences and as such the research needed to ascertain that the students knew that the items were referring to the mathematical software used in their course. Thus, the covering letter that was sent to the students indicated the course, the software and in what context learning mathematics was considered. Further, changes were made from referring to teacher to tutor with keeping with the Open University terms. A text copy of the questionnaire is attached along with a covering letter (see Appendix 2, p.285).

Included with these two inventories was an open section where students were encouraged to type in comments on learning mathematics with software. Background information such as gender and age were obtained from the students’ administrative records.

4.2.2 Sample Profile

In all, 388 replies (22%) were obtained for the online questionnaire of which 371 had completed all parts. The lowest response was from the T207 (9%) and MST121 (13%) courses. The highest response rate was from D841 (36%). Of the 388 respondents, 34% of the respondents were from courses with statistical software, 26% from courses with CAS and 42% from courses with spreadsheets.

There were slightly more male respondents (55%) than females. The ages of the respondents were divided into 3 groups, young (<30), middle-age (30 to <50) and old

(≥50). About 67% of the respondents were between the ages of 30 to 50, whilst only 16% each in the less than 30 and more than 50 age groups. The gender distribution varied with age (χ^2 (2) = 6.83, p = 0.03), that is, the men tended to be older than the women in this sample.

4.2.3 Factor Analysis of Two Inventories

To determine whether mathematics confidence was related to processing levels, a factor analysis was performed. Firstly, since the ALMQ was scored oppositely to the MCAS, the deep and surface scores were recoded to calculate their scores such that a low score reflected a higher deep or surface processing level. The ALMQ was chosen to be recoded rather than the MCAS because there were fewer scales (2 versus 6) and this minimised any arising researcher error such as arithmetic and calculation errors. The full list of items for the Mathematics Engagement scale was used in the questionnaire but as suggested by Galbraith and Haines (2000b) two items were excluded when calculating the scores. Cronbach's coefficient alpha was used to test the reliability of the items. The Mathematics Engagement scale had a low reliability (0.38) compared to the other scales (see Table 13). A correlation analysis showed that all scales were significantly correlated with each other except for Mathematics Engagement which did not correlate with any of the other scales. It was hence removed from the factor analysis.

A principal component analysis using a varimax orthogonal rotation with a Kaiser normalization extraction method was performed. Two factors were identified that had eigenvalues greater than 1 (Kaiser in Field, 2000), two factors were extracted that accounted for 65% of the variance in the data. An oblimin oblique rotation was also performed to determine whether the oblique rotation or orthogonal rotation was appropriate for the data. Field (2000) explains an oblique rotation is used when the extracted factors are related whilst an orthogonal rotation is used when the extracted factors are independent. To test whether the extracted factors were related, the factor correlations were examined. As the factor correlations for the oblique rotation were low (0.38), the orthogonal rotation was used (Field, 2000; p.439). The results of the varimax rotation performed in SPSS are presented in Table 13.

Table 13: Factor analysis of scores from the MCAS and ALMQ

	Factor 1	Factor 2	Coefficient alpha
Mathematics Confidence	**0.89**	0.12	0.92
Mathematics Motivation	**0.90**	0.21	0.88
Computer Confidence	0.36	**0.76**	0.86
Computer Motivation	0.13	**0.93**	0.83
Computer-Mathematics Interaction	0.11	**0.91**	0.74
Surface	**-0.72**	-0.11	0.79
Deep	**0.73**	0.23	0.90

Thus, this factor analysis agreed with Galbraith and Haines' (2000b) findings in that the computer attitude scales were distinct from the mathematics attitude scales. Further, these results were in agreement that the Computer-Mathematics Interaction scale was related to students' Computer Confidence and not their mathematics attitudes. Also, whilst Galbraith and Haines found that Mathematics Engagement was associated with Mathematics Confidence and Mathematics Motivation, these results showed that it had no association with confidence and motivation.

What was of more interest was that this factor analysis showed that Mathematics Confidence loaded positively for the Deep processing level and negatively for the Surface processing level on Factor 1. This suggests that these scales are related to each other and thus the conjecture that Mathematics Confidence and the processing levels were related confirms the findings of Duff (2004) that academic self-confidence and the deep processing level were correlated using the RASI.

Thus, any of the studies that had to be designed had to account for students' mathematics confidence when looking at students' understanding of mathematics as this could have impacted on their processing levels.

4.3 Pilot Study 2: Expected Values

The Expected Values pilot study was intended for testing the remote observation method. Also this study was used to test whether software boxes could be programmed in Excel and used for collecting relevant data with respect to the three task types.

There were four parts in testing the remote observation process:

- Determine the mathematical domain and tasks
- Develop the software boxes
- Determine the experimental design
- Test and set up the remote observation equipment

As indicated in Section 3.4.1 (p.66), a mathematical topic was chosen that was most likely unfamiliar to undergraduate students. Thus, a problem from the operations research/management science domain was chosen as this is mostly taught at the university level but required only secondary school level mathematics. The topic of expected values from decision theory was selected as it was considered a simple concept that students could understand in approximately twenty minutes. This topic used probabilities to compare and determine the best option from various alternative options. The probability concept is usually learnt by students at the secondary level.

4.3.1 Design of Study

Instructional materials on expected values and the principles for calculations were derived from Winston (1994). Whilst the expected values' options were referred to lotteries in decision theory, when developing the instructional materials, this word was replaced with the word ‘game’ to minimise confusion as the constructive tasks referred to real-life applications such as insurance. Expected values tasks were chosen and modified from Winston (1994) to fit the taxonomy of Galbraith and Haines (2000a). For the mechanical and interpretive tasks, there were three games each that students had to determine the best option.

With respect to software boxes, there was no known software package that was exclusively used to solve expected values and, had there been one, it would probably have been of a black-box type. Thus, the three software boxes were developed in Excel spreadsheet using visual basic application (VBA). The black-box application allowed students to calculate the expected values without showing steps whilst the glass-box software performed calculations showing the steps for each game. Open-box software allowed the students to interact with the software at each step for the game particularly with respect to choosing the arithmetic calculation and in so doing calculating the answer. All three software boxes were developed in separate Excel sheets (see Figure 10 on p.103 and Figure 11 on p.104). There were three more sheets also developed: 1) the data entry sheet in which the data could be entered before using the software boxes, 2) the scrap sheet in which students could do calculations if they wish and 3) the answer sheet in which students recorded their answers during the post test (see attached CD for examples of the software boxes).

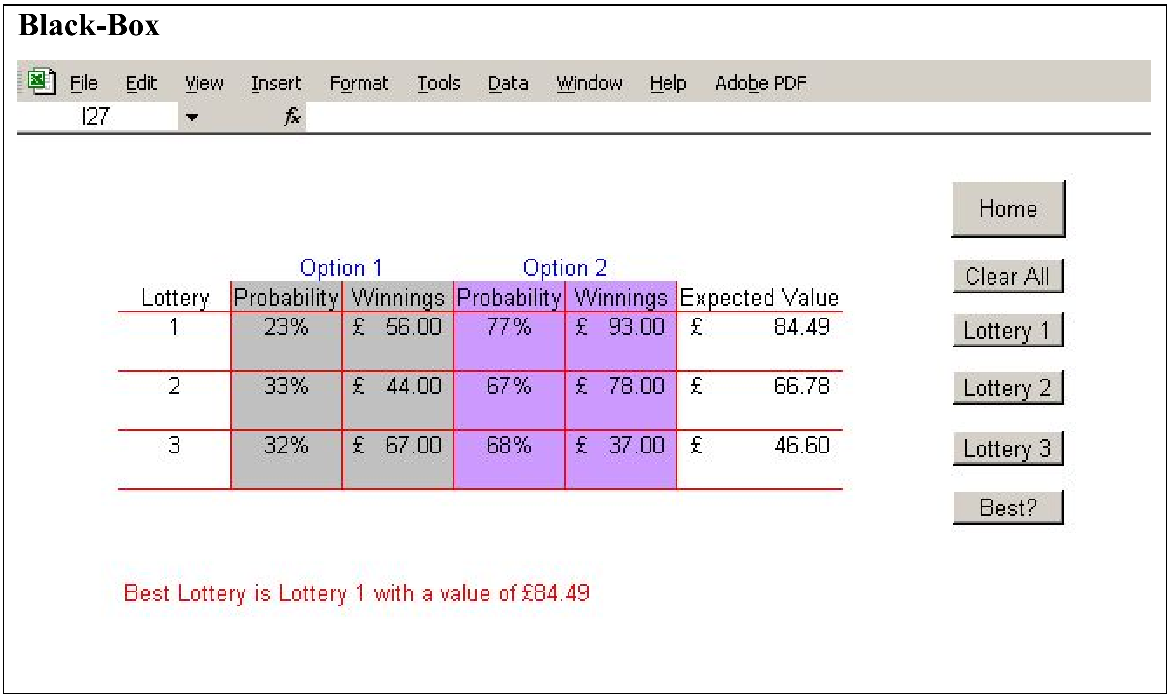


Figure 10: Screenshot of using the black-box software for solving expected values

As indicated in Section 3.3.2 (p.59) an experimental protocol used by Renkl *et al.* (2004) was employed. The background questionnaire was set up as a web-page and sought to find answers relating to age, level of mathematics acquired, gender, confidence in mathematics, computers and Excel spreadsheet, and whether they had any knowledge on expected values. These values were intended to be used as covariates in the main study analysis. Whilst the recommendation for any covariates, particularly ones where the participants self-identified their levels should have a series of questions (e.g. Owen and Froman, 2005), it was felt that subjecting participants to a longer series of items would require more time from students and may affect the number of students willing to participate in the study. The instructional materials as mentioned before provided background information on expected value and were also supplemented with a technical manual on how the three software boxes worked (see Appendix 3, p.294).

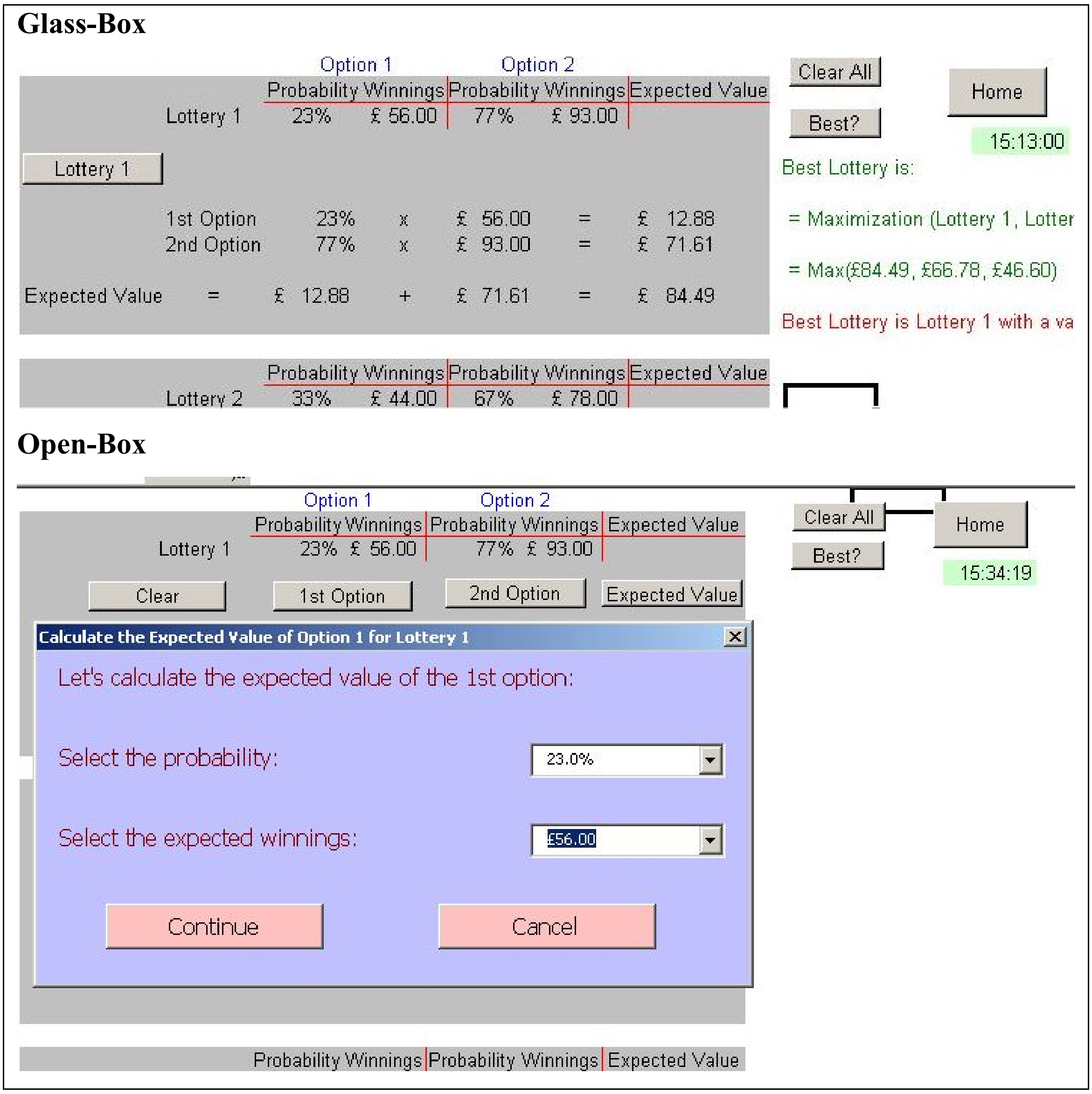


Figure 11: Screenshots of using the glass and open-box software for solving expected values

The pre-test consisted of using simple probability questions which were awarded one mark each (see Appendix 3, p.294). An example of a pre-test question is as follows:

> *If a dice is rolled, what is the probability that the dice will have a value of four or more?*

The pre-test was based solely on simple probability since Renkl *et al.* (2004) suggested using a level of difficulty that was lower than the post-test.

In the post-test, there were 9 tasks (three from each task type): the first 6 tasks were multiple-choice and related to the mechanical and interpretive tasks (see Table 14). The last three tasks (constructive) required the participant to solve by pen-and-paper or the software box and to provide an explanation for the answer. The answer sheet in Excel was used for entering the answers for the post-test.

Table 14: Examples of expected value tasks

Mechanical Task: Which of the following games would I get the best expected value for? Game 1: 1st prize: 47% probability of winning £105 2nd prize: Expected prize of winning £58 Game 2: 1st prize: Expected prize £98 2nd prize: 37% probability of winning £129 Game 3: 1st prize: 78% probability of winning £68 2nd prize: Expected prize of winning £135
Interpretive Task: Which of the following games would I get the best expected value? *r* is an arbitrary probability. Give your reasoning. Game 1: 1st prize: (*r*-30%) probability of winning £56 2nd prize: Expected prize of £25 Game 2: 1st prize: *r* probability of winning £55 2nd prize: Expected prize of £25 Game 3: 1st prize: (*r* + 10%) probability of winning £25 2nd prize: Expected prize £21
Constructive Task: Joan's assets consist of £10,000 in cash and a £90,000 home. During a given year, there is a 0.001 chance that Joan's home will be destroyed by fire or other causes. How much would Joan be willing to pay for an insurance policy that would replace her home if it was destroyed?

Following the post-test a short interview was conducted with the students to elicit their opinions on the three software boxes and on expected values. Each task was awarded one point each. An additional practice task was also provided which was mechanical in nature. All nine tasks were unrelated to each other. Only for the mechanical tasks, students were expected to solve using the software. For the interpretive tasks, these were expected to be logically deduced.

This study used 6 students for testing the remote observation process and a rotational confounded study design (Campbell and Stanley, 1963) was tested, where each student used the three software boxes in 6 permutations. Students solved all mechanical tasks first, followed by the interpretive and the constructive tasks. The students solved one task type each with one software box (see Table 15).

Table 15: The sequence the software boxes were used to solve expected values tasks

Student	Tasks 1, 4, 7	Tasks 2, 5, 8	Tasks 3, 6, 9
1 B	Open	Black	Glass
2 J	Glass	Open	Black
3 G	Open	Glass	Black
4 Cl	Glass	Black	Open
5 Ch	Black	Glass	Open
6 R	Black	Open	Glass

4.3.2 Remote Observation: Testing

Using the remote observation protocol for this study, students were expected to have either MSN messenger or Windows messenger installed on their computers and working on a Windows platform. Further, the students were required to have a web camera and headphones with a microphone. These were necessary for the observation of the students. Students used the remote application facility on their computer to connect to the researcher's computer where they are able to interact and use the software boxes (see Figure 12).

Netmeeting was the software required for using the application-sharing facility on a Windows platform and it is pre-installed on Windows. Application-sharing allows a user (Researcher) to share a software application (MS Excel) via the internet to another user (Student). The receiving user (Student) is able to see the MS Excel application and request and receive control from the sharing user (Researcher) to use the software application.

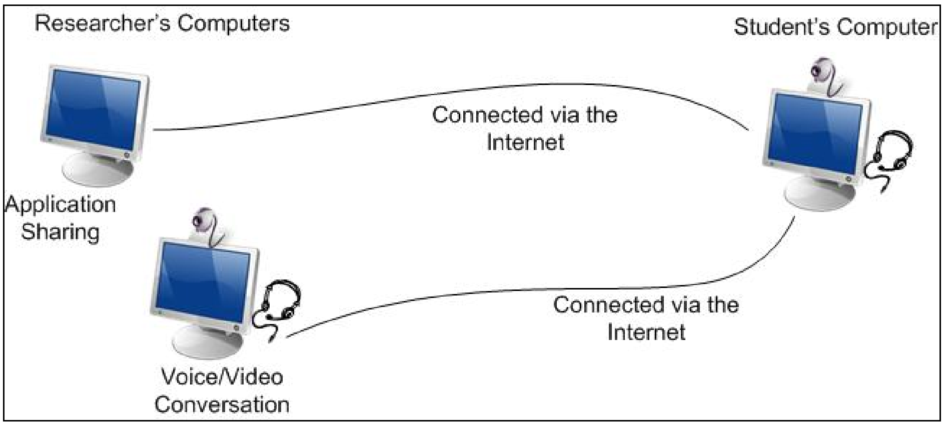


Figure 12: An illustration of the remote observation process

Through the students' web-cameras and video conversation facilities available on Windows Messenger, students were observed and interviewed whilst using the software. By using screen and audio capture software, students' on-screen actions, web camera video and audio were all recorded.

Before the actual observation date, students were contacted via email on whether they would like to participate in the study. These students were Open University post-graduate students. The email indicated that they would be audio and video recorded and advisory times for the session (see Table 16). Although initially students were asked to load Netmeeting prior to the study, this actually occurred during the remote observation session as it only took a couple of minutes to load.

Following this email with students agreeing to take part in the study, a time and date was set up. Students were then emailed two web-links. The first web-link was to a consent form on the webpage. The consent form for students participating in the remote observation study was challenging as signed consent was difficult to obtain when students were at a distance. Students were required to fill in their names on the form but this provided no guarantee that this was indeed the student filling in the form. Therefore, to remedy this situation during the actual experimentation period students were asked for permission again as to whether they consented to be video and audio recorded and if there were any objections. The second web-link was linked to a webpage that had the background questionnaire. These web-links were used to minimize experimental time required by the student and provided more flexibility.

Table 16: Sample of request email

Dear ________ For the remote observation study, I need persons who have at least done mathematics at the secondary/ high school level. I'll be video-taping (via a webcam) you whilst you use a software (Excel) on the computer (which will also be recorded i.e. your actions on the computer via a screen capture software). Your voice conversation with me via the computer will also be recorded. As such you'll need to have a webcam and headphones loaded on your computer before the study. You will also need to have installed windows Netmeeting onto your computer which comes with Windows and also Windows Messenger or Msn Messenger installed. I'm hoping the remote observation session doesn't take more than 1 1/2 to 2 hrs. I may want to follow up our session with at an interview at some other time. The remote observation will require you to do the following (the times are just guidelines - you may take less or more time): 1. Fill in a survey (5 mins) 2. Read Introductory materials into the topic (5 mins) 3. Do a pre-test (10 mins) 4. Read through the software (Excel) materials instructions (10 mins) 5. Have a play with the software (15 mins) 6. Practice the talk-a-loud strategy (5 mins) 7. Do the post-test (40 mins) Thanks for helping. Anesa

Two to three days before the remote observation session, students were emailed the instructional materials, the practice question and the post-test questions. All three materials were held in one Word document. This allowed time for the students to print these materials and have it ready on hand for the remote observation session. They were discouraged from reading the materials until the remote observation session. The intention was to minimise students preparing or learning the topic prior to using the software boxes.

The pre-test questionnaire link was provided to the student via Windows Messenger and was filled in during the experimental session. They were also told that it was not necessary to read these materials prior to the experiment. During the experiment, students were given time to read through the instructional materials on expected values and the software materials. The students were instructed to use a practice task for testing the three software boxes and also to practice the think-aloud strategy which constituted step 4 of the experimental protocol (Table 6, p.60).

4.3.3 Results and Implications

Students indicated that they eventually forgot about being seen with the web camera as when they maximized the screen with the Excel application this window went to the back. This perhaps improved the observation process as there was less sense of feeling that they were being watched. This however did not mean that any Hawthorne effect had been removed completely (see Landsberger, 1958). The Hawthorne effect is where students may work harder on tasks in response to being observed. The students also had the convenience of using their own computer and environment, so they were aware of where applications were located and where they could find implements such as pen, paper or calculators.

However, some students indicated that they often felt a break in concentration when they were prompted to keep talking. Further, it appeared that most speech (and possibly more explanations) occurred during the practising of the mechanical task and thus for the rest of the tasks there were few explanations. Most participants eventually said what they were doing felt like the same thing being repeated for both the tasks (mainly for the mechanical and interpretive tasks) and the software boxes. For example, Student 1B said the following when she was doing the tasks in the following sequence of software boxes

> Open-box: *"Same calculations … same as the first one" …"They're all the same"*
>
> Black-box: *"It's quicker, but all the same"*
>
> Glass-box: *"Different layouts for them, but all the same"*

Student 1B in her reference to open-box software was indicating that the tasks were all similar. When it came to doing the tasks using the black-box software, she mentioned that the black-box software was quicker and the glass-box software had a different layout, but essentially they were all the same when it came to solving the tasks.

The expected values tasks were perhaps quite simplistic as most students made passing comments to the effect that it would have been faster to do it by hand (for example Student 2J) and whether they could use pen and paper instead (e.g. Student 6R). Student 5Ch however was the opposite in that initially he did not want to do the calculations using pen-and-paper as he did not understand how to calculate the expected values. Afterwards using the different software boxes, he then preferred to solve the tasks using pen-and-paper as he indicated that he had now learnt how to do the tasks by watching the software boxes. All students were able to obtain full marks on the mechanical tasks. Further, as the same instructional materials were used by all students this meant that there was no privileging of prior teaching styles (Kendal and Stacey, 2001) and therefore this would not influence the way they learnt.

The interpretive tasks required mainly conceptual knowledge, in which students could have worked out mentally by using mathematical logic for the solution. However, some students opted to use trial-and-exploration methods of testing values either through the software boxes, or using algebraic equations to solve for the unknown quantity, *r*, although *r* was not required to be solved. This showed that for the interpretive tasks, students brought in other knowledge such as the algebraic models but were also likely to resort to the software boxes by exploring and testing their hypotheses. For example, in the interpretive task, some students worked out that $r \leq 90\%$ and $r \geq 20\ \%$, and tested values for *r* in this range, until they could conclude which game was better. Also, students appeared to explore with the black-box or the glass-box software more than the open-box software as the latter required more interactivity.

For the interpretive tasks, there appeared to be more self-explanations occurring for the black-box and glass-box than for the open-box software. Whilst for the constructive tasks, the students using the open-box and glass-box software appeared to have more self-explanations. However, this may be the nature of the tasks (being of a contextual nature) rather than the software boxes itself. Thus, in the Main Study there was a need to find out whether it was the task type that was causing the use of the software boxes in that way, that is, if there was a relationship between the task types and the software boxes used, or whether it was just the tasks alone that elicit that type of reaction.

Also, there was a likelihood that open-box software promoted self-explanations more than the others since it had prompts. The open-box software was also considered tedious by the students, and perhaps the reason for avoiding it during exploration. The possible reason for this was that it perhaps needed a higher cognitive effort than the others, the procedures overly simplistic or it was just badly designed as students often mentioned that there was an irritating pop-up box.

4.4 Pilot Study 3: Linear Programming

Based on the expected values study, it was noted that students thought the tasks were quite simple and resorted to using pen-and-paper. This meant that a more complicated mathematical domain was needed that students could not easily work out with pen-and-paper in order to determine the influence of the software and steps. Further reasoning for choosing linear programming was also indicated in Section 3.4.1 (p.66). However, choosing the actual aspect of linear programming was challenging. The aim of this pilot study was to test whether the linear programming domain could adequately be used as tasks for testing the three software boxes.

There were three main parts in linear programming: the formulation of the problem, the solution to the problem and the sensitivity analysis. When it came to steps, the research focused on the solution part requiring the simplex method as this was part of the task that could be easily converted into a type of software box, although the sensitivity analysis could do the same; this would have required more complex linear programming concepts such as the duality of the problem. The research needed to keep the introduction of new concepts to the students at a minimum to ensure there was not a large cognitive effort required. The dual problem required a more complex simplex method (two phase simplex method) which would be beyond the student to learn or understand in their first introduction to linear programming.

In using the simplex method to solve linear programming problems, there are several steps that the user may have to do (Winston, 1994):

1.Convert the problem into canonical form

2.Decide what are the basic variables

3.Decide the entering variable

4.Calculate the ratio

5.Decide the pivot row

6.Perform elementary row operations

Whilst all these steps were needed, perhaps the key steps were from Step 3 to Step 5, as these required some ‘rule of thumb’ to do. For example, at Step 3 deciding the entering variable indicated which variable should be increased to provide the largest profit, whilst calculating the ratio and deciding the pivot row indicated how much the variable can be increased without violating the constraints (see also Section 3.7, p.86).

The formulating of a word problem may or may not add to conceptual and procedural knowledge but this could not be translated easily to the software boxes, although perhaps it was able to add context to the students. As such the proposal for the tasks was worded problems that were already formulated for the students. An added advantage of using already formulated problems was that this ensured all students starting from the same point rather than having to account for wrongly formulated problems. Further, as the formulation would most likely occur through pen-and-paper, this type of data would be lost or obscured through the remote observation method.

4.4.1 Design of Study

Thus only the simplex algorithm which solved the problem using linear algebraic methods was considered. These were again developed in Excel spreadsheets using VBA, because although several linear programming software packages were examined (e.g. Excel Solver, MathLab, Lindo and MathCad), the software packages were unable to demonstrate the abilities of the black, glass and open-box software. Figure 13 represents a schema of the steps required for developing the three software boxes. Two options were considered for the open-box software (represented by OB in the figure). In the expected values pilot, the students had to do several arithmetic operations for the open-box software, the choice was in this case to allow the student to do only one operation per step as it minimised the cognitive effort required by the students. Hence the choice of choosing the appropriate pivot variable was selected as the step (OB current). The steps for black-box and glass-box software are represented by BB and GB respectively in the figure.

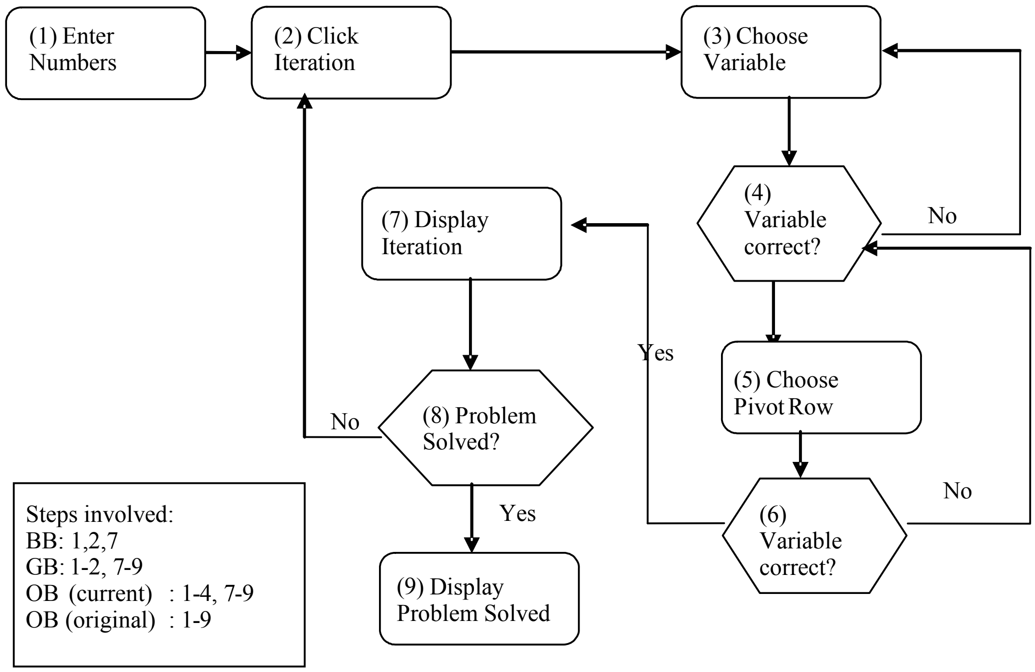


Figure 13: The schema for developing the black-box, glass-box and open-box software

Snapshots of the black-box software (Figure 14), the inputting of the values (Figure 15), the choice of pivot variable for the open-box software (Figure 16) and the iterations and solutions for both glass-box and open-box software (Figure 17 and Figure 18) are shown below. An annotated screen shot (Figure 7, p.73) along with the explanations of all the buttons was discussed in Section 3.4.3 (p.71).

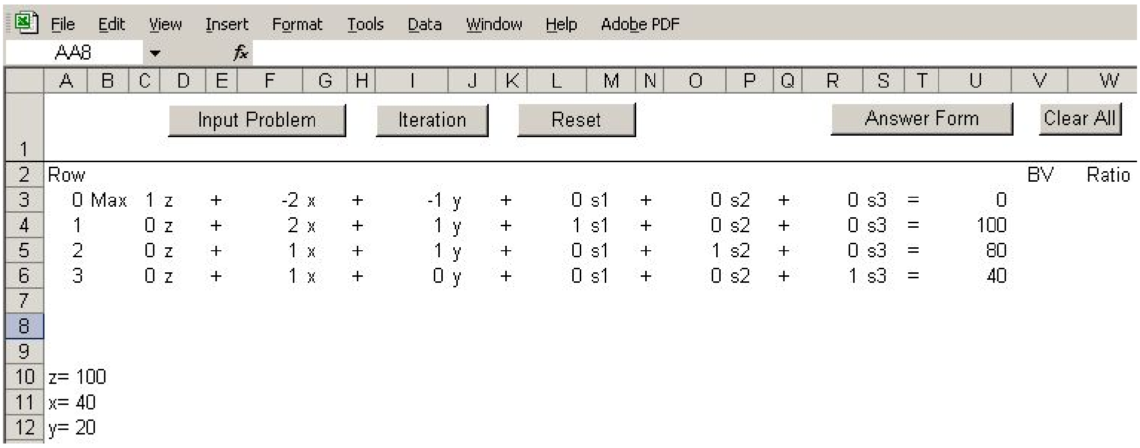


Figure 14: Linear programming black-box software showing the canonical form and solution

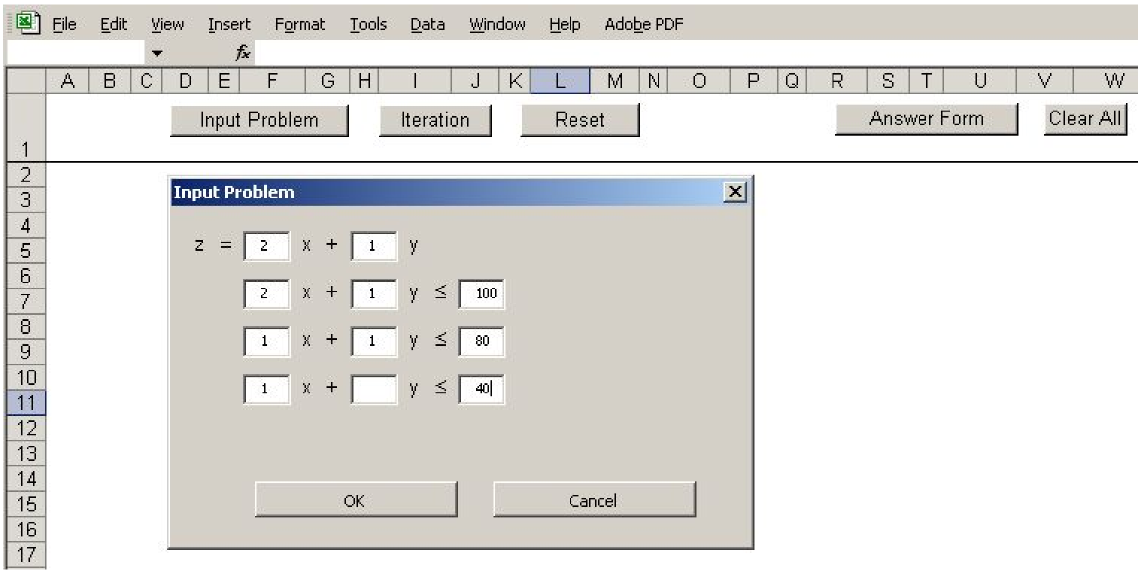


Figure 15: Linear programming software-box showing input problem screen

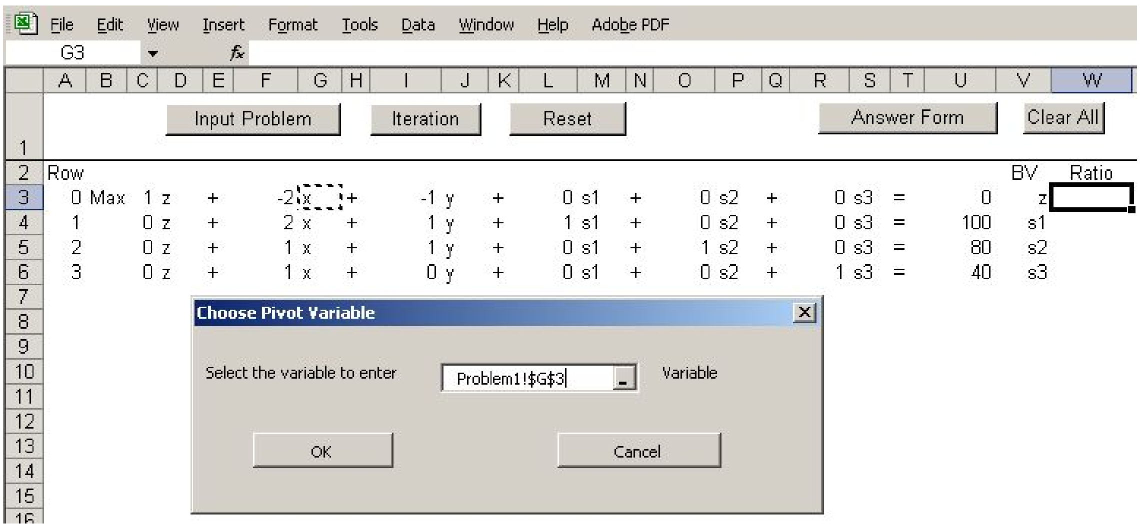
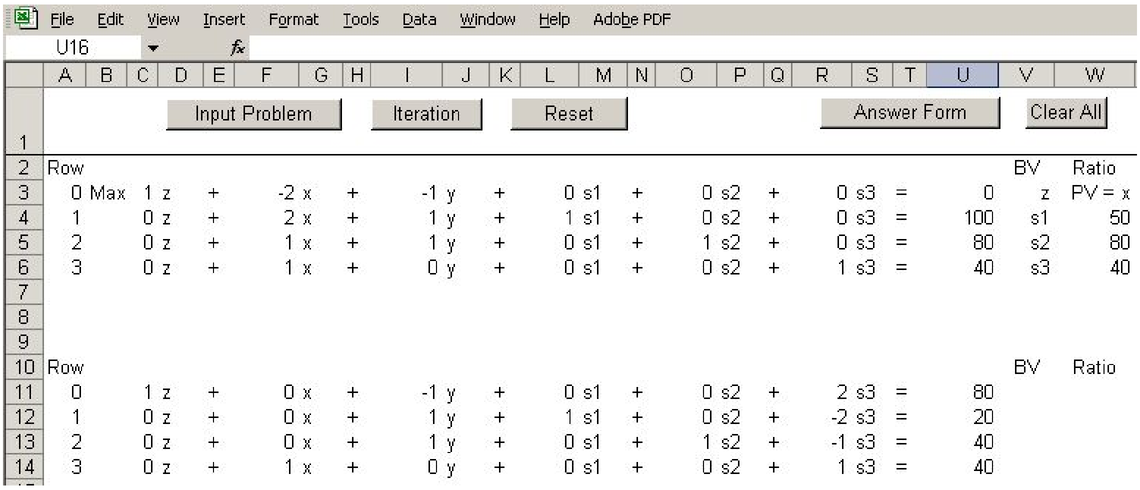


Figure 16: Linear programming open-box software showing the choosing of the pivot variable



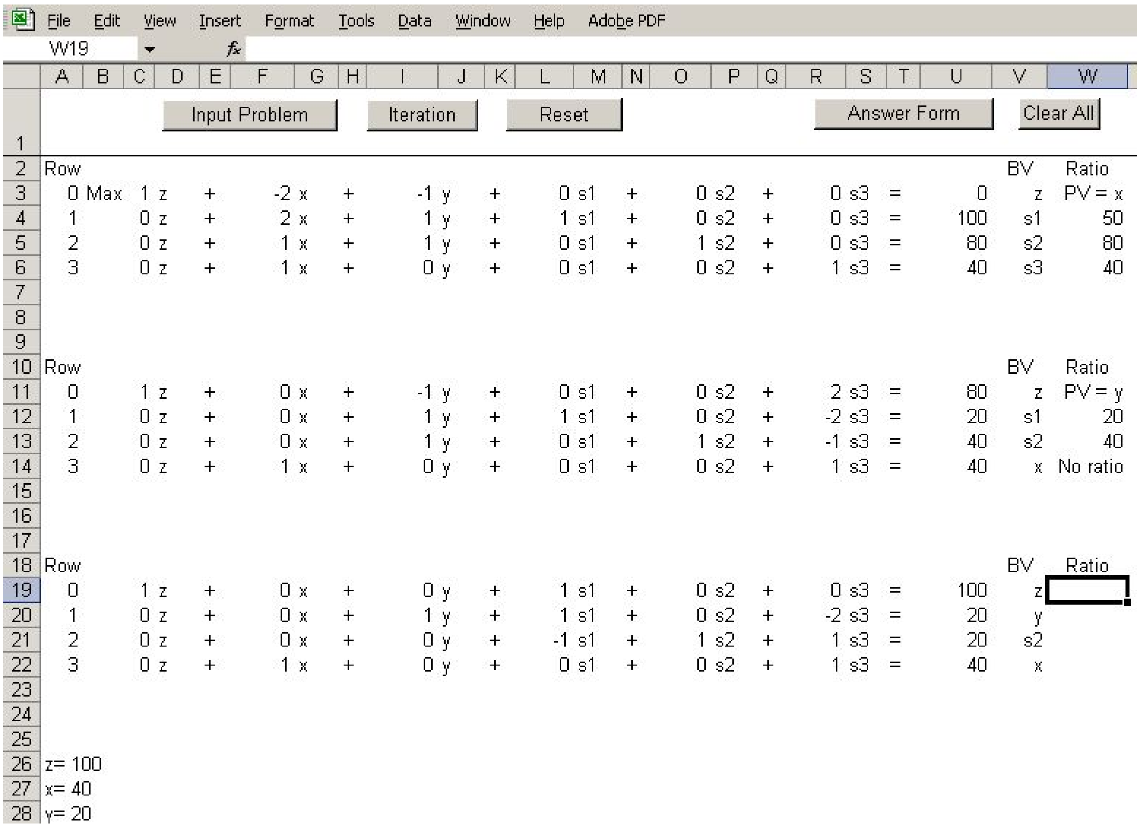
File Edit View Insert Format Tools Data Window Help Adobe PDF

U16

Input Problem | Iteration | Reset | Answer Form | Clear All

	Row	z		x		y		s1		s2		s3		RHS	BV	Ratio
2	Row														BV	Ratio
3	0 Max	1 z	+	-2 x	+	-1 y	+	0 s1	+	0 s2	+	0 s3	=	0	z	PV = x
4	1	0 z	+	2 x	+	1 y	+	1 s1	+	0 s2	+	0 s3	=	100	s1	50
5	2	0 z	+	1 x	+	1 y	+	0 s1	+	1 s2	+	0 s3	=	80	s2	80
6	3	0 z	+	1 x	+	0 y	+	0 s1	+	0 s2	+	1 s3	=	40	s3	40
10	Row														BV	Ratio
11	0	1 z	+	0 x	+	-1 y	+	0 s1	+	0 s2	+	2 s3	=	80		
12	1	0 z	+	0 x	+	1 y	+	1 s1	+	0 s2	+	-2 s3	=	20		
13	2	0 z	+	0 x	+	1 y	+	0 s1	+	1 s2	+	-1 s3	=	40		
14	3	0 z	+	1 x	+	0 y	+	0 s1	+	0 s2	+	1 s3	=	40		

Figure 17: Linear programming software (glass-box or open-box) showing first iteration



File Edit View Insert Format Tools Data Window Help Adobe PDF

W19

Input Problem | Iteration | Reset | Answer Form | Clear All

	Row	z		x		y		s1		s2		s3		RHS	BV	Ratio
2	Row														BV	Ratio
3	0 Max	1 z	+	-2 x	+	-1 y	+	0 s1	+	0 s2	+	0 s3	=	0	z	PV = x
4	1	0 z	+	2 x	+	1 y	+	1 s1	+	0 s2	+	0 s3	=	100	s1	50
5	2	0 z	+	1 x	+	1 y	+	0 s1	+	1 s2	+	0 s3	=	80	s2	80
6	3	0 z	+	1 x	+	0 y	+	0 s1	+	0 s2	+	1 s3	=	40	s3	40
10	Row														BV	Ratio
11	0	1 z	+	0 x	+	-1 y	+	0 s1	+	0 s2	+	2 s3	=	80	z	PV = y
12	1	0 z	+	0 x	+	1 y	+	1 s1	+	0 s2	+	-2 s3	=	20	s1	20
13	2	0 z	+	0 x	+	1 y	+	0 s1	+	1 s2	+	-1 s3	=	40	s2	40
14	3	0 z	+	1 x	+	0 y	+	0 s1	+	0 s2	+	1 s3	=	40	x	No ratio
18	Row														BV	Ratio
19	0	1 z	+	0 x	+	0 y	+	1 s1	+	0 s2	+	0 s3	=	100	z	
20	1	0 z	+	0 x	+	1 y	+	1 s1	+	0 s2	+	-2 s3	=	20	y	
21	2	0 z	+	0 x	+	0 y	+	-1 s1	+	1 s2	+	1 s3	=	20	s2	
22	3	0 z	+	1 x	+	0 y	+	0 s1	+	0 s2	+	1 s3	=	40	x	
26	z= 100															
27	x= 40															
28	y= 20															

Figure 18: Linear programming software box (glass-box or open-box) showing iterations and solution

The study design was again similar to that for the expected values, but in this case students were only expected to use one software box. This ensured that students did not feel as if the software boxes were repetitious and also to decrease any fatigue that students may acquire through the learning of three different software boxes. Three problems were given but this time each problem had three parts, where each part was a mechanical, interpretive and constructive task (see Table 17).

These three tasks were all related to each other. Both the interpretive and constructive tasks were dependent on the mechanical task. This was used to diminish the feeling that all the tasks were the same and ensured students would not feel as if they were repeating the same task again. Students were thus required to compute the mechanical task correctly in order to do the interpretive and constructive tasks.

Table 17: Example of a linear programming problem

Linear Programming Problem: a) Solve Max $2x_1 + x_2$ $2x_1 + x_2 \leq 100$ (constraint A) $X_1 + x_2 \leq 80$ (constraint B) $X_1 \leq 40$(constraint C) $X_1, x_2 \geq 0$(Mechanical Task: 2 marks) b) If x_1 = no. of toy trains manufactured and x_2 refers to the no. of toy soldiers manufactured, and constraint A refers to painting hours, constraint B to carpentry hours and constraint C, the demand for toy trains. Interpret what this solution means to the toy company who wants to maximize their profit by producing toy trains and toy soldiers. Provide as detail answer as possible. (Interpretive Task: 2 marks) c) If the cost of trains has increased by £0.50, how would this affect the number of toy trains and toy soldiers being sold? Provide as detail as an answer as possible. (Constructive Task: 2 marks)

In the expected values study, the students solved all the mechanical tasks correctly because it only required the students inputting the values and clicking calculate. Even in the open-box software where students had to interact with steps in the expected values study, the students were still able to choose the appropriate steps. Therefore for the linear programming mechanical tasks, the same was expected, that is, all students will calculate the mechanical task correctly. This was particularly true in the linear programming open-box software, in that, as students only had to choose the correct pivot variable, this would not affect the computation in any way. There was also the possibility that the students would get the wrong answer if they inputted values incorrectly for any of the software boxes. Thus, the researcher ensured that the students entered the correct values and by doing this, it meant that students were able to achieve the same performance scores for the mechanical tasks regardless of software boxes.

The three problems were selected; two were of application types and the other was an abstract type. The first application problem dealt with the manufacturing of toys and the second with the manufacturing of furniture. Further, the abstract problem and the application problems were mixed in to determine whether more explanations were occurring for the application problems versus the abstract problem. However, this would only be an indication since there was only a comparison between three problems.

Only for the mechanical tasks were students required to use the software boxes. Students could have answered the constructive task with or without software, that is, it was possible to solve the constructive task using pen-and-paper. It was expected that for the interpretive task, that students would not use the software boxes. The software boxes were used only to show the procedural steps and hence it was expected from examining the software boxes that students will build their procedural knowledge. If indeed there is a conceptual-procedural link as suggested by Rittle-Johnson *et al.* (2001) (see Sections 1.3, p.3 and 2.3.1, p.19) then students' conceptual knowledge should be impacted on when they examine the procedural steps.

However, as indicated in Section 2.3.3 (p.21), for students to build any conceptual knowledge, it would depend on the approach (that is their level of engagement such as using a relational understanding approach) that students undertake when examining not only the procedural steps from the software boxes but also the instructional materials. Thus, if students engage with the procedural steps from the simplex algorithm, they may notice that the slack variables' values are also calculated. Conceptually a student may realise that a calculated slack value would mean that there is a surplus of resources for that constraint. The constructive tasks are devised to take advantage of these calculated slack values, that is, in all of the constructive tasks, the students are asked what will occur when a constraint with surplus resources was increased.

Therefore, if the students were able to engage with the procedural steps and also build conceptual knowledge, they probably would not need the software box to recalculate the constructive task but instead determine the answer from examining the linear programming problem and its calculated answer (from the mechanical task). If they were unable to build this conceptual knowledge, then recalculation, that is using the software box, would be their only option. Further a difference in interpretive task scores for the software boxes may also indicate that students were able to build conceptual knowledge from the software boxes, as the interpretive tasks mostly require the application of conceptual knowledge.

Since the interpretive and constructive tasks were dependent on the answer from the mechanical task, it was imperative that the students got this correct and hence the researcher ensured that numbers entered were correct. Further, the software boxes were devised to indicate to the student when the best solution was found. In this Pilot Study, three students participated to test one software box each (that is Student 1 tested the black-box, Student 2 the glass-box and Student 3 the open-box). The remote observation process occurred similarly to that of the expected values study.

4.4.2 Results and Discussion

The data collected from the student using the glass-box software was unfortunately lost because of computer hard-drive problems due to insufficient space during video rendering. However, the data from the other two students using the black-box and the open-box were transcribed and coded. During the mechanical and interpretive tasks students did not provide numerous out-loud self-explanations. The student using the open-box software was more talkative and seemed keen on understanding how the iterations within the open-box software worked. It was uncertain whether this was because of the type of student he was or it was genuinely the open-box software encouraging this type of reflection. Students provided self-explanations for the black-box and open-box software in all tasks (see Table 18 and Table 19 for explanations in the constructive task).

Table 18: Self-explanations for the constructive questions using the black-box

"Okay then let's have a little think" : Thinking about why *"Well, I expect it'll reduce demand maybe … increase profit? … oh well, let's have a guess"*: Coming to an explanation (Real-life) *"Well, obviously it would increase something … it would increase the amount of furniture … I suppose … I don't know how much … and which bits"*: Thinking about why (Real-life)

Table 19: Self-explanations for the constructive questions using the open-box

"You need to find a relationship between the cost … the increase in cost and the demand … can you do it?" : Thinking about why (Mathematical) *"Obviously to get the same maximum profit, you need less trains, the x would have to be lower"*: Coming to an explanation (Mathematical) *"You would expect an increase in cost to reduce the demand, but if you didn't you will have … more profit"*: Relating to life *"What I can't understand, there is no relationship between cost and demand in the equations*": Relating to mathematics principles

Whilst the student using the black-box software did not have many explanations the student using the open-box software did; again, it was uncertain if this was because of the type of student. The open-box software student made explanations in relation to the mathematics principles whilst the black-box software student made explanations which tended to be towards real-life. Thus, the dividing of the explanations into mathematical and real-life seemed to work in this situation particularly where there were application problems.

4.5 Implications for the main study

Linear programming seemed to work reasonably well in allowing students to use the software boxes. Further the developed tasks did not appear to have the same issue as in the expected values of having the feeling of ‘sameness’.

Using money with decimal places in the linear programming study seemed to give students more difficulty than rounded figures and as such tasks were changed to have rounded figures in case there were students who found decimal figures challenging. Also, as there were hardware problems this meant that equipment with larger storage capacity that could handle the data was also needed.

4.5.1 Updates from Pilot Study 3 to the Main Study

There were a few changes in how the data were collected from Pilot Study 3. First of all, students were allowed to use either Skype or Windows Live Messenger for the voice/ video conversation. Further, if students used Skype then an additional programme, Unyte Application Sharing was used for sharing the Excel spreadsheet. This meant that students using other operating systems such as computers operating on a Mac operating system were also able to participate. Students using Windows Live Messenger continued to use the application sharing facility that came in-built with Windows.

Further, students were expected to be recruited through the internet such as through advertisements on Facebook and on web forums however only three students were recruited. Of these three, only one took part in the study. Instead, students were recruited through gatekeepers from Trinidad and Tobago. The gatekeepers installed computer set-ups in two locations, one using Windows Live Messenger and the other using Skype. Skype was used instead because Local Area Network (LAN) permissions would not allow the use of Windows Live Messenger voice/ video conversation permission. Since the gatekeepers recruited the students for only one session, the background questionnaire and consent form were given to the students at the start of the session to fill in before the observation session started.

Windows Live Messenger provided the best results for the remote observation, since Skype sometimes had conflicts with Unyte application sharing which caused it to be unstable and in some cases resulting in the lost of data. Further, Skype sends voice/video data at 32 kbps (Windows Live Messenger: 16 kbps) bandwidth, and so faster speeds were required; as a result, smooth conversation was not always achieved with Skype since the internet speeds in Trinidad and Tobago were lower than in the UK.

The new equipment that was used provided a slightly updated method to the remote observation method where only one computer by the researcher was used which used large screen to capture the Excel spreadsheet and the web camera videos (see Figure 19 compared with Figure 12, p.107).

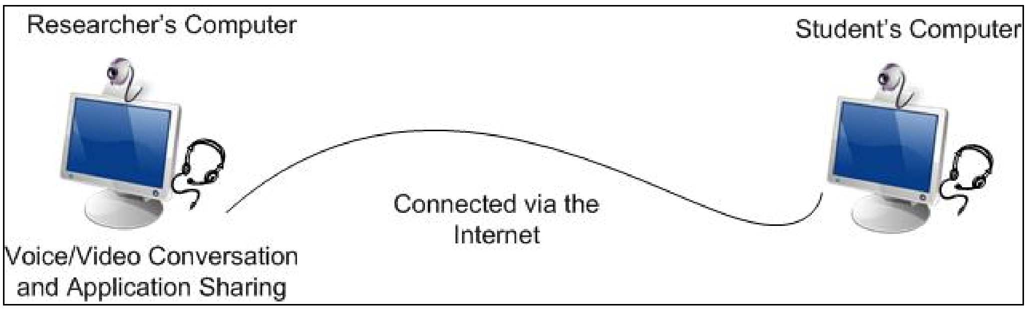


Figure 19: Updated remote observation method

An external hard-drive of 320 GB was used to store the video data for all of the 38 students which ensured that the computer's hard-drive had sufficient space for storing students' data that were being recorded at that time. The random access memory was expanded to 1 GB to accommodate for the rendering of video. Also, a dedicated Asymmetric Digital Subscriber Line (ADSL) was used instead of the LAN as minimum LAN security was needed in sending and receiving video and doing application sharing.

4.6 Concluding Remarks

From Supporting Study 1, it was shown that mathematics confidence was related to deep processing level and thus confirmed the findings from Duff (2004) (Section 4.2, p.95). Further, from the two pilot studies (Pilot Study 2 and Pilot Study 3), it was confirmed that the remote observation could be used for observing students and collecting useful data for purposes of answering the research questions. Further, it was noted from Pilot Study 2 that simple tasks were not well suited for testing the software boxes and hence a more complicated mathematical domain was needed which could force the students to use the software boxes rather than pen-and-paper (Section 4.3, p.101). There was more success in ensuring that students used the software boxes when linear programming was used in Pilot Study 3 (Section 4.4, p.112).

Further from the Pilot Studies, the three software boxes were coded and developed in MS Excel as well as tested in two mathematical domains. From these pilots, linear programming was chosen as the mathematical topic for the Main Study. MS Excel appeared to be a reasonable program for configuring the three software boxes and also stable for conducting the remote observation studies as there were no particular complaints of using it. This is probably because these students were already familiar with it as it is one of the programs used during secondary school and university.

Chapter 5. Main Study: Quantitative Data Analysis

"No more than these machines need the mathematician know what he does."
- Jules Henri Poincaré

5.1 Introduction

This chapter reports on the quantitative findings from the Main Study. The Main Study used an experimental design in which students answered three problems each having three associated tasks: mechanical, interpretive and constructive. Two problems were application-oriented and one was abstract. Each student was randomly assigned to one of the three software boxes.

Based on the remote observation method developed and discussed in Chapter 4, data collected were: a) observational data of the students through the use of web cameras and screen capture software, b) typewritten answers to tasks and c) audio data.

This chapter investigates quantitatively the following three areas:

1. If students' performance varies across: problems, tasks and software boxes
2. If the identified approaches in Chapter 2 (explorations, explanations and processing levels) vary with problems, task and software boxes and
3. If there is any relationship between the identified approaches and performance and if the relationship varies with the software boxes.

This chapter begins with providing a sample profile of the participants that took part in the study (Section 5.2, p.125). There are a number of statistical terms and tests presented in this chapter. These variables and statistical terms are briefly explained in Section 5.3 (p.131). The reliability of the scoring of marks by the research and the coding of explanations are discussed in Section 5.4 (p.136). The performance scores of the students across software boxes and tasks are then investigated (Section 5.5, p.139) which is followed up with further investigation of students using the exploration approach (Section 5.6, p.146), the explanation approach (Section 5.7, p.152) and the processing levels approach (Section 5.8, p.161) for the software boxes and tasks. The chapter rounds up with a discussion on how the analysis in this chapter relates to the research questions (Section 5.9, p.165) and finally concluding remarks are made (Section 5.10, p.172).

5.2 Sample Profile

This section provides an overview of the participants used in the analysis. The analysis was performed on data collected from 38 students.

5.2.1 Participants in the Data Collection

In all 46 students were recruited for the data collection but only 38 students were used. The reasons for this are outlined in this section. After the data were collected for Participant 23 (glass-box), Participant 38 (glass-box) was used to replace her because the former did not use the spreadsheets to solve the mechanical tasks. Instead, she chose to solve the mechanical tasks by pen and paper. This meant that she did not arrive at the correct answer for the mechanical tasks, unlike all the other students who used the software. Having the correct answer for the mechanical task was essential for solving the interpretive and constructive tasks (Section 4.4.1, p.113), thus she was removed from further analysis.

Whilst recording the observation session with Participant 5 (black-box), the audio data were corrupted and subsequently Participant 37 (black-box) was used as a replacement. This ensured that data from the think-aloud self-explanations were obtained. Participant 39 (glass-box) was given the wrong sequence of problems and hence another student had to be recruited. The typewritten answers from both Participants 5 and 39 were included in the quantitative analysis.

Subsequently after the data collection, during the synchronization of video and audio, it was found that the audio data for Participant 13 (glass-box), Participant 18 (glass-box) and Participant 26 (open-box) were missing or corrupted. The reason for this seemed to be a mixture of poor internet connection, the use of Skype and a conflict with the Unyte Application Sharing software. Skype uses a higher quality of audio; it sends data at 32 kilobytes per second as opposed to MSN messenger which sends audio data at 16 kilobytes per second. This meant more internet bandwidth was being occupied by Skype, and this provided choppy audio. In some of the audio recordings, before Unyte Application Sharing software started, both the researcher and the participant audio were recorded. However, once the Unyte Application Sharing software started, the participant's audio stopped recording perhaps because of a software conflict. Therefore, transcribing these participants think-aloud sessions was not considered.

In addition, two participants recruited from Facebook (both from the UK) did not participate in the study. For one participant, the researcher was unable to receive video data due to recent changes to the wireless network permissions at The Open University, which blocked the transmission of video data. The other participant was interested in taking part but was unable to obtain a web camera although she indicated that she should be receiving one soon. Although sending a web camera to her was considered (but not communicated to her), she then indicated she had other commitments. It should be noted that both of these students did not reach the stage of being assigned a participant identification number.

Similarly, two other participants who had volunteered withdrew from taking part in the experiment because of other commitments. One participant was from the UK and was suggested by Participant 9, the other was recruited by a gatekeeper. Three more participants started the experiment but two sessions were cancelled because of poor internet connections. Poor internet connection caused the software application to run slower such that calculations of the software boxes on the internet-shared Excel file were slow and voice conversations were choppy. This was frustrating to the students and the researcher. In the third case, there was a conflict between Skype and Unyte Application sharing which caused the computer to crash and thus all data were lost.

All 38 students' typewritten answers were marked and coded into real-life and mathematical explanations. A summary profile of the final 38 participants with their participation identification numbers is provided in Appendix 4 (p.302).

5.2.2 Gender, Age, Mathematics Level and Degree

Of the 38 participants, 36 participants were under 25. The percentage of males (47%) and females (53%) were fairly similar (see Table 20). Most students had attained A-level mathematics (50%) or had done some type of university mathematics (32%), with the remaining students acquiring General Certificate of Secondary Education (GCSE) equivalent mathematics (18%).

Students were categorised into three discipline groups: physical sciences (physics/ mathematics/ engineering), life sciences (biology/ biochemistry/ medicine) and other. There were twice as many students from the life sciences than from the physical sciences. Two gatekeepers were primarily from medicine and biology and the third gatekeeper was mainly from physics or mathematics which possibly influenced the disciplines from which the students were drawn. Only 3 students were categorised into 'other' discipline (economics, tourism, general BSc). There was a gender difference ($\chi^2(2) = 7.01$, $p = 0.03$) in that there were more males than females within the physical sciences and more females than males in the life sciences. The physical sciences usually attract more males than females and this may be the reason for this distribution.

Table 20: Distribution of participants across discipline, gender and mathematics level

Mathematics Level	Discipline			
	Phys. Sci.	Life Sci.	Other	Total
Female				
GCSE	0	3	0	**3**
A-level	1	8	1	**10**
University	2	3	2	**7**
All	**3**	**14**	**3**	**20**
Male				
GCSE	0	4	0	**4**
A-level	4	5	0	**9**
University	5	0	0	**5**
All	**9**	**9**	**0**	**18**
Total	**12**	**23**	**3**	**38**

Students performed similarly in the pre-test regardless of mathematics level, gender or disciplines with an average mark of 4.3 (*SD* = 1.2) out of 6. Most of the students (34) did not know the basic principles of linear programming. Indeed, many (14) had never even heard of linear programming. Only four students indicated that they had previously solved any linear programming problems. Further examination showed that two of the students (Participants 3 and 8) had completed mathematics only up to the Advanced Level (A-level) or equivalent. Thus, they possibly solved the linear programming problems within the A-level curriculum. These A-level problems are not related to the simplex algorithm which is used in this research but rather the graphical method. The remaining two students (Participants 5 and 29) completed mathematics at the university level and they were both in Engineering (Physical sciences). For these two students, it was possible that they had done some advanced linear programming. However their scores did not indicate that they were at any advantage to the other students as their total scores from the post-test were within the confidence limits.

5.2.3 Mathematics, Excel and Computer Confidence

Students assessed their own confidence levels on a ten-point scale on doing mathematics, using computers and using Excel (where 1 = low and 10 = high). Students had a mean mathematics confidence of 6.2 (*SD* =1.70) (see Table 21).

Table 21: Mean confidence scores on mathematics, computer use and excel use by mathematics level, gender and discipline

	N	Mean Confidence Scores		
		Mathematics	Computer	Excel
Mathematics Level				
GCSE/ equivalent	7	4.4	7.4	6.0
A-Level	19	6.7	7.4	6.5
University	12	6.6	6.3	5.4
Gender				
Female	20	5.8	6.2	5.6
Male	18	6.8	8.1	6.6
Disciplines				
Physical Sciences	12	7.1	7.0	6.0
Life Sciences	23	6.0	7.3	6.1
Other	3	5.3	5.3	6.0
All	38	6.3	7.1	6.1

The females' self-assessment was significantly lower than males for mathematics confidence ($F(1,36) = 4.23$, $p = 0.05$), computer confidence ($F(1,36) = 13.84$, $p < 0.01$) and marginally significant for Excel confidence ($F(1,36) = 3.95$, $p = 0.06$). The difference in mathematics confidence is possibly an effect of mathematics level. Females with a General Certificate of Secondary Education (GCSE) mathematics level had a lower mathematics confidence (2.7) than males with GCSE (5.8). As their GCSE grade was not recorded, it is uncertain whether their grade level was an influence. Note that regardless of mathematics confidence, both genders scored similarly in the pre-test (males: 4.1 and females: 4.6).

Although the difference in the computer confidence was about 2 points between males and females; their difference in Excel confidence was about 1 point which meant that the lack of computer confidence for the females may not be strongly related to the use of MS Excel.

Whilst students were confident in using computers ($M = 7.1$, $SD = 1.8$), they were less confident in using Excel ($M = 6.0$, $SD = 1.7$). Across the different disciplines, students had similar computer and Excel confidence. Students who had only GCSEs were significantly less confident in mathematics ($F(2,35) = 6.66$, $p < 0.01$) than the students who had done A-level or university mathematics.

5.2.4 Distribution of Students for Software Boxes and Sequence

Although students were randomly assigned to a Sequence (that is, a sequence of problems), chi-square tests were performed to determine whether students were distributed equally across groups. Any uneven distribution was due to chance rather than design since the students were assigned randomly. The distribution of students by pre-test scores, mathematics level, gender, disciplines, mathematics confidence, computer confidence and Excel confidence across the three software boxes was similar. Hence, the students were evenly distributed based on their ability, confidences and gender in their assigned software groups.

Students were also randomly assigned to which sequence they would answer the problems. Students in their respective sequence was evenly distributed for discipline ($\chi^2(4) = 4.49$, $p = 0.34$) and gender ($\chi^2(2) = 1.57$, $p = 0.46$). However, students who had done university mathematics (75%) were more likely to answer problems in Sequence 2 ($\chi^2(4) = 13.84$, $p < 0.01$).

Students who solved problems in Sequence 3 had the least mathematics confidence ($M = 4.92$, $SD = 1.68$) ($F(2,35) = 8.62$, $p < 0.01$). Students' distributions of Excel confidence ($F(2,35) = 0.04$, $p = 0.96$) and computer confidence ($F(2,35) = 0.05$, $p = 0.95$) were similar across the Sequences. Students' total pre-test scores were also found to be similarly distributed across sequence ($F(2,35) = 0.21$, $p = 0.81$).

5.3 Understanding the Variables and Statistics

This chapter uses statistical analysis to address the first two research questions and to some extent the third research question. Using the first research question as an illustration, the variables and statistical terms are explained.

The first research question was interested in whether performance scores were dependent on the software boxes used. Performance scores consisted of marks received from interpretive and constructive tasks within each problem. Mechanical task scores were not included since the researcher ensured that all students got this task correct by drawing the students' attention to mistakes in input. All tasks and problems were answered by each student; hence these two variables were manipulated within-subjects. Scores were obtained for three problems and six tasks (three interpretive and three constructive); this meant that for each student there were three scores associated with the variable Problem and 6 scores associated with the variable Task. Throughout this chapter, when referring to a variable in a statistical analysis, the variable's first letter is capitalised. For example, 'Problem' and 'Task' are referring to the variables in the statistical design and have a value associated with them. The software Boxes were the group variable and hence manipulated between-subjects.

To determine whether there was an influence of software Boxes on performance, a mixed-design analysis of variance (ANOVA) with both between-subjects and within-subjects factors was conducted. The means obtained or reported in this chapter for any variable is based on this ANOVA which provides an average of all means. For example, as each task was scored out of two, this meant that the combined score of Task was the mean of the interpretive and constructive scores, which meant that Task means scores were out of two as well. Further, the means for Problems were the mean of Task means across all problems and thus the reported mean Problem scores were also out of two.

The Latin-Square experimental design required analysing two variables, Sequence and Question for determining if there was any carry-over effect by solving problems in different sequences. The Sequence group variable measured the scores obtained based on the sequence in which problems were presented to the student (Section 3.3.5, p.63), for example Sequence 1 represented students solving Problem 3 first followed by Problem 1 and then Problem 2. ‘Question’ is used to differentiate from Problem and represents which problem the student is answering for example, Question 1 represented students solving their first question such as Problem 1 if in Sequence 3, Problem 2 if in Sequence 2 or Problem if in Sequence 1. An ANOVA indicated that Sequence and Question did not influence performance scores and these variables were hence removed in other ANOVAs involving performance scores (Annex 3, p.324). Note that the statistical annexes are all located under Appendix 7 (p.315).

5.3.1 ANOVA Assumptions and Statistical Terms

The ANOVAs performed in this study, the scores for the Problems and Tasks did not have a normal distribution. This was probably due to the relatively small number of students doing these tasks. Although, the assumption of normality for the ANOVA has been violated, the ANOVA is robust for moderate departures from the normal distribution (Howell, 2002: p.323). Further, Rider and also Pearson in Glass, Peckham and Sanders (1972) found that non-normality on the *F*-tests were not affected providing the degrees of freedom of the residual variance were not too small. In this study, the residual variance degrees of freedom varied from 29 to 64 and these degrees of freedom may be reasonable.

The within variables of Problems and Tasks were also found to have heterogeneous variance based on Levene's test of homogeneity. Schultz (1985) explained that the Levene's test is conservative for small sample sizes and it is perhaps why heterogeneity was observed for the small sample sizes used in this study. Thus, the ANOVA was considered sufficiently robust against this heterogeneity and was used in this study (see Annex 1, p.316 for a more detailed explanation).

The concept of effect sizes (Cohen, 1988) is also used within this thesis. An effect size represented by eta-squared (η^2) is used to understand how large a difference is found when a significant difference ($p < 0.05$) is obtained in an ANOVA. The effect size measures the proportion of variance in the dependent variable, that is explained by the independent variable.

In this thesis, instead of quoting effect size in the ANOVAs, partial effect size (η_p^2) is used. According to Tabachnick and Fidell (2007), η_p^2 should be quoted instead of η^2 when there are several variables. In the ANOVAs, the variables used are Problem, Task, Mathematics Confidence and software Boxes. Partial effect size is the proportion of variance explained by the independent variable, partialling out the effects of the other independent variables and any interactions between them. It is represented mathematically as the proportion of the effect variance (SS_{effect}) to its variance and its associated error variance (SS_{error}):

$$\eta_p^2 = \frac{SS_{effect}}{SS_{effect} + SS_{error}}$$

A consequence of using η_p^2 is that the effect sizes do not add up to one unlike η^2. Cohen (1988) provided guidelines for sizes of η^2 but not for η_p^2. Richardson (through personal communication as reported by Ramanau, 2007) explained that Cohen put forward criteria for small, medium and large values of eta-squared, but that he himself applied this to values obtained when partialling out the effects of covariates. Hence, they can also be used when partialling out the effects of other independent variables, as with partial eta-squared. Thus the guidelines used for η_p^2 are the same used for η^2, which were that 0.01, 0.06 and 0.14 represented 'small', 'medium' and 'large' effects, respectively. A more detailed explanation on effect sizes including Richardson's explanations is provided in Annex 1 (p.316).

Further, when an interaction is observed for example between Task and Box, Fisher's Least Significant Difference (LSD) test was used for post-hoc analysis. The LSD test takes into account the confidence limits to determine where the differences lie (see Annex 1, p.316 for more information).

5.3.2 Mathematics Confidence – Introduction of a Grouping Variable

An initial ANOVA using a model of Box, Problem and Task for predicting performance scores showed that there was no influence of software Box on performance scores. However, upon adding Mathematics Confidence (MathConf) as a covariate through an analysis of covariance (ANCOVA), there appeared to be a Task by Box interaction. This meant that MathConf was influencing the Task by Box interaction. However, the interaction effect was being adjusted by the covariate rather than a main effect. When an interaction effect is adjusted, it means that the variable (MathConf) is causing a violation of the homogeneity of slopes assumption in the ANCOVA.

The usual practice for resolving this violation caused by the MathConf variable, is to split the variable and recode it. Within the psychology discipline, the way to split a continuous variable, in this case the Mathematics Confidence covariate, is to dichotomise the variable using either the mean or median split. Whilst a number of researchers advise against splitting a variable (see Cohen, 1983; Maxwell and Delaney, 1993; Owen and Froman, 2005), Owen and Froman indicated that when there is a homogeneity of slopes violation in a repeated measures ANCOVA, splitting the covariate maybe the most legitimate course of action. A full explanation of the violation and its resolution is provided in Annex 2 (p.320).

The mean for the Mathematics Confidence distribution was 6.3 and the median at 6.5 (see Figure 20), thus either the median or mean can be used for the split, as 6 would be the demarcation point. Mathematics Confidence was thus recoded into lower and higher Mathematics Confidence. This variable was called MathConfRec.

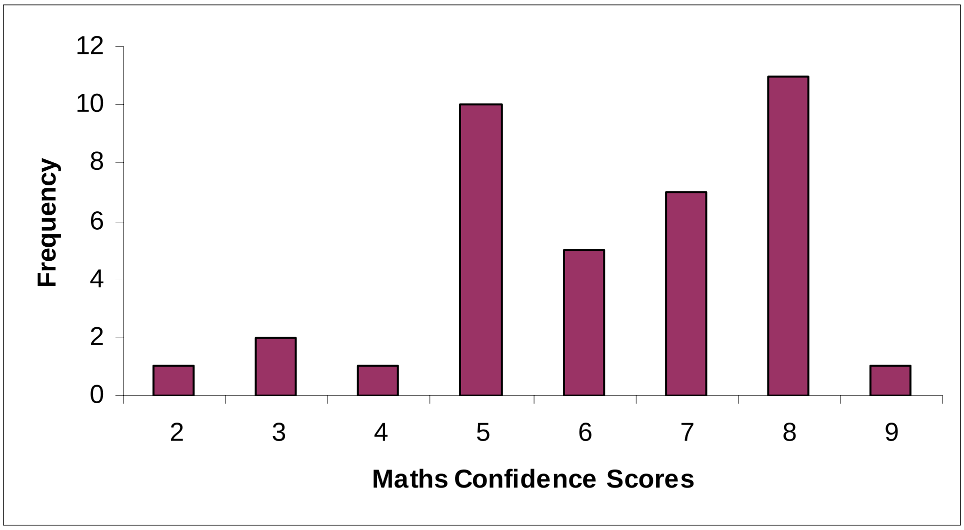


Figure 20: Distribution of frequency scores for Mathematics Confidence

Lower Mathematics Confidence was grouped for the range 2 to 6 and higher Mathematics Confidence for the range 7 to 10. This evenly divided the students into groups of 19 participants. This split makes sense in that they appears to be two peaks one at 5 and one at 8. This suggests that there are probably two levels of Mathematics Confidence.

Note that the students in this study generally had assessed their performance quite highly with 34 of the 38 students assessing themselves 5 and over. Whilst the groupings were divided into lower and higher, it was with respect to this group of students. The lower confidence group may not genuinely represent students who have very low mathematics confidence. The reason why these students probably had these high mathematics confidence scores were that they were from disciplines that had mathematics elements and that 80% of the students had learnt mathematics at either A-level or university.

The sub-groups of Box and Mathematics Confidence did not have equal sample size (see Table 22), although a chi-square suggested that they were similarly distributed ($\chi^2(2) = 3.85$, $p \leq 0.15$). From hereon, when referring to Mathematics Confidence as the variable, this represents the recoded Mathematics Confidence, MathConfRec.

Table 22: Distribution of students based on the software Box and Mathematics Confidence

Box	Mathematics Confidence		Total
	Lower	Higher	
Black	4	9	**13**
Glass	9	4	**13**
Open	6	6	**12**
Total	**19**	**19**	**38**

In this thesis, when reporting values for Mathematics Confidence based on MathConfRec, the terms Higher MC and Lower MC are sometimes used to represent Higher and Lower Mathematics Confidence respectively. Also, the actual Mathematics Confidence score is provided when reporting on a particular participant.

5.4 Reliability Analysis of Marks and Explanations Coding

The answers from the students were scored based on a marking scheme which allocated 2 marks each to the interpretive and constructive tasks. Both the interpretive and the constructive tasks had generally two parts, one part which required the students to write an answer to the question and another part which asked the student to explain why they chose this answer (see Appendix 6, p.313). These two parts were given one mark each.

Further, each interpretive and constructive task was coded into whether the explanations provided were real-life or mathematical explanations. A coding scheme was developed for the mathematical and real explanations. Mathematical explanations were easier to define than real-life explanations. For mathematical explanations the following should hold true, the student:

- Wrote a mathematical equation e.g. $2x + 3y = 20$ which aided in an argument
- Used inequalities or equalities to help the explanation e.g. $x <= 20$; $y = 5$; x is greater than y. That is they must say something like "Since $x <= 20$ and $y = 5$ then ….". Thus, an argument must ensue from writing it in this way
- Calculated a value or some indication of calculating a value e.g. z will be negative; 40 - 20 = 20; x approaches infinity; no change to x and y
- Did not rewrite numbers that were calculated by the software e.g. number of chairs produced = 20
- Explored numbers (not given in the mechanical task) with the software and typed the solution from their exploration or indicated they had explored and also gave an explanation as above

The symbol <= is used to represent to ≤ in the explanations above as this was the way that students inputted the sign during the session since ≤ is a formatted symbol and not a keyboard input button.

The real-life explanations were more obscure and seemed to draw from life-experiences. Real-life explanations were:

- Not based on any explicit mathematics such as numbers or equations but using rules of thumb or heuristics that were commonplace such as "increasing production would increase profit"

- When students drew conclusions based not on any mathematics but rather on what they ‘felt’ might be true e.g. “the demand is too low to have any production”
- When students drew from their own feelings or experience on how the manufacturing world works e.g. “this would not be economical”.

Examples of both of these explanations are presented in Table 23.

Table 23: Types of explanations provided by students for a constructive problem

Participant 4 (F, BB, Higher MC = 8): *Want to produce more trains if the profit is increased.* (Real-Life Explanation) Participant 9 (F, BB, Higher MC = 7): *profit would increase to 140 but the numbers of toys made stays the same because constraints is that x =40 maximum so even though they get more profit they can’t make any more trains*. (Mathematical Explanation resulting from an exploration) Participant 13 (M, GB, Higher MC = 8): *If the profit per train is increased, it would likely be more profitable to produce more trains and fewer soldiers. However Constraint C puts an upper bound on the number of trains that can be produced -- a bound which has already been achieved. Hence it is not possible to produce more trains and the number of toy trains and toy soldiers produced would remain the same. This was confirmed by solving the modified problem*. (A Mathematical Explanation confirmed with the testing of the software)

The answer transcripts from all 38 students were marked and coded according to the marking and coding scheme by one researcher. To ensure that the marking and coding were consistent, five randomly selected answer transcripts were coded by another judge and the reliability between these two judges were determined. These transcripts were from Participants 7, 8, 20, 24, and 25. Participant 15’s transcript was used as an example for showing how the answer scripts were marked and coded by the principal researcher. Inter-rater reliability between the judges for the scores and coding were calculated based on the intra-class correlation (ICC) using a 2-way random Analysis of Variance (ANOVA) design (see Shrout and Fleiss, 1979).

Upon discussion between the two judges about the marks and coding, the ICC for performance scores was calculated as 0.96, whilst real-life and mathematical explanations were 0.94 and 0.73 respectively. Inter-rater agreement was determined between the two judges based on the percentage of similarity between scores and explanations. The inter-rater agreement for scores (83%) was lower than the real-life (97%) and mathematical explanations (87%). The observed disparity between the inter-rater agreement and inter-rater reliability for performance scores was because performance scores allowed half marks and hence whilst it may correlate between two judges as near equal, it meant that this showed disparity in full agreement. The low inter-rater reliability and inter-rater agreement of the mathematics explanations as opposed to the real-life explanations and scores was possibly due to the second judge being unfamiliar with linear programming and not primarily from a mathematical domain.

5.5 Performance Scores

The purpose of this section is to determine if performance scores on Tasks were affected by Boxes. To accomplish this, an ANOVA was performed (see Table 24). As indicated in Section 5.3 (p.131), the Sequence and Order variables were removed because they had no influence on scores. Also, Mathematics Confidence was added as a recoded variable (Section 5.3.2, p.134).

5.5.1 Performance on Problems and Tasks

From the ANOVA, the performance scores on Tasks were dependent on software Boxes as the Task × Box interaction was significant. Before proceeding into understanding this interaction, this section first provides an overview of the students' performance on Problems and Tasks and then looks at the influence of the software Boxes.

Table 24: ANOVA of Box×Mathematics Confidence×Problem×Task

	SS	df	MS	F	p	η_p^2
Between Subjects						
Box	0.06	2	0.03	0.04	0.96	0.00
MathConfRec	2.25	1	2.25	3.42	0.07	0.10
Box × MathConfRec	0.42	2	0.21	0.32	0.73	0.02
SS within	21.10	32	0.66			
Within Subjects						
Problem	12.58	2	6.29	29.41	0.00	0.48
Problem × Box	1.11	4	0.28	1.30	0.28	0.08
Problem × MathConfRec	2.15	2	1.07	5.02	0.01	0.14
Problem × Box × MathConfRec	0.65	4	0.16	0.76	0.56	0.05
Problem × SS within	13.68	64	0.21			
Task	17.04	1	17.04	78.11	0.00	0.71
Task × Box	2.84	2	1.42	6.51	0.00	0.29
Task × MathConfRec	2.00	1	2.00	9.15	0.00	0.22
Task × Box × MathConfRec	0.23	2	0.11	0.53	0.60	0.03
Task × SS within	6.98	32	0.22			
Problem × Task	3.58	2	1.79	6.70	0.00	0.17
Problem × Task × Box	1.58	4	0.40	1.48	0.22	0.08
Problem × Task × MathConfRec	1.33	2	0.66	2.48	0.09	0.07
Problem × Task × Box × MathConfRec	1.69	4	0.42	1.58	0.19	0.09
Problem × Task × SS within	17.11	64	0.27			

In all three problems, the students performed the best in Problem 2 (1.04) but attained about half this mark in Problem 1 (0.48) and Problem 3 (0.55). The reason for this difference is uncertain as all three problems had the same format (Section 3.4.2, p.68), that is, they each had a mechanical, interpretive and constructive task. The interpretive task in all three problems required the student to examine the solution computed during the mechanical task and to then interpret what the solution meant within a particular situation.

Although the interpretive task in Problem 3 had the same format, there was an extra part which asked students about a general condition of linear programming. Similarly, all the constructive tasks required understanding what happened if a value was changed in the linear programming model. They either explained this through examination of the task or solving the linear programming problem again. Therefore the problem formats were perhaps not the reason for the difference in scores. The problem type, that is, being abstract or applied, is also not a reason for the observed difference since Problems 1 and 2 are both application problems but students performed significantly better in Problem 2 than Problem 1.

The problem scores comprised the mean of the interpretive and constructive task scores. By examining these tasks individually, this will shed light on why there is a difference in the Problem scores. Overall students performed better in the interpretive task (0.98) than in the constructive task (0.40). As mentioned in Section 5.3 (p.131), the Task scores provided are means and are all out of two. Note that the sum of the Task mean scores will not be equal to the sum of the Problem scores, however the mean of all Problem scores will equal to the mean of all Task scores.

Students performed similarly in the interpretive task for Problem 2 (1.23) and Problem 3 (1.03). Figure 21 illustrates these results. Students' mean score for Problem 1's interpretive task (0.68) was at least 0.5 marks lower than that of Problem 2. Students scored the highest in Problem 2's constructive task (0.84) but did abysmally in the same task for Problem 1 (0.29) and Problem 3 (0.07).

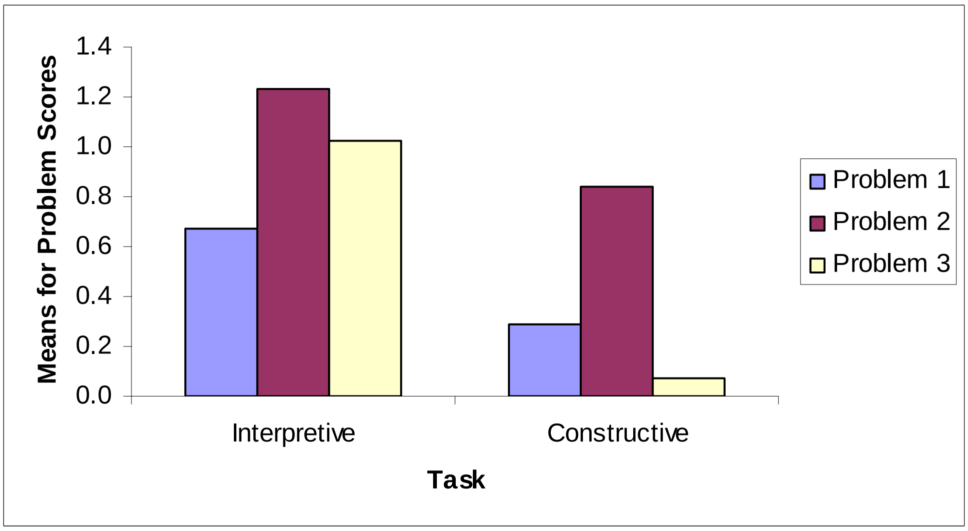


Figure 21: Means scores for Task depending on the Problem

There are two things that are highlighted here, firstly that students performed well in Problem 2 regardless of which task they were given, that is, they scored highly in both the interpretive and constructive tasks. Problem 2 was perhaps an 'easy' question for the students and why this was easy will be further discussed in Chapter 6. An interesting phenomenon of Problem 2 was that students performed similarly regardless of their Mathematics Confidence level (Higher MC: 1.00 vs Lower MC 1.08). This was not the case for Problem 1 and Problem 3 where higher mathematics confidence students scored higher in these problems than the lower mathematics confidence students (see Figure 22).

Generally across all problems, students with higher mathematics confidence (0.79) did marginally better ($p = 0.07$) than students with lower mathematics confidence (0.58). Whilst mathematics confidence influenced interpretive task scores (Higher MC: 1.18 vs Lower MC 0.70), there was no apparent influence on the constructive task scores (Higher MC: 0.41 vs Lower MC: 0.40).

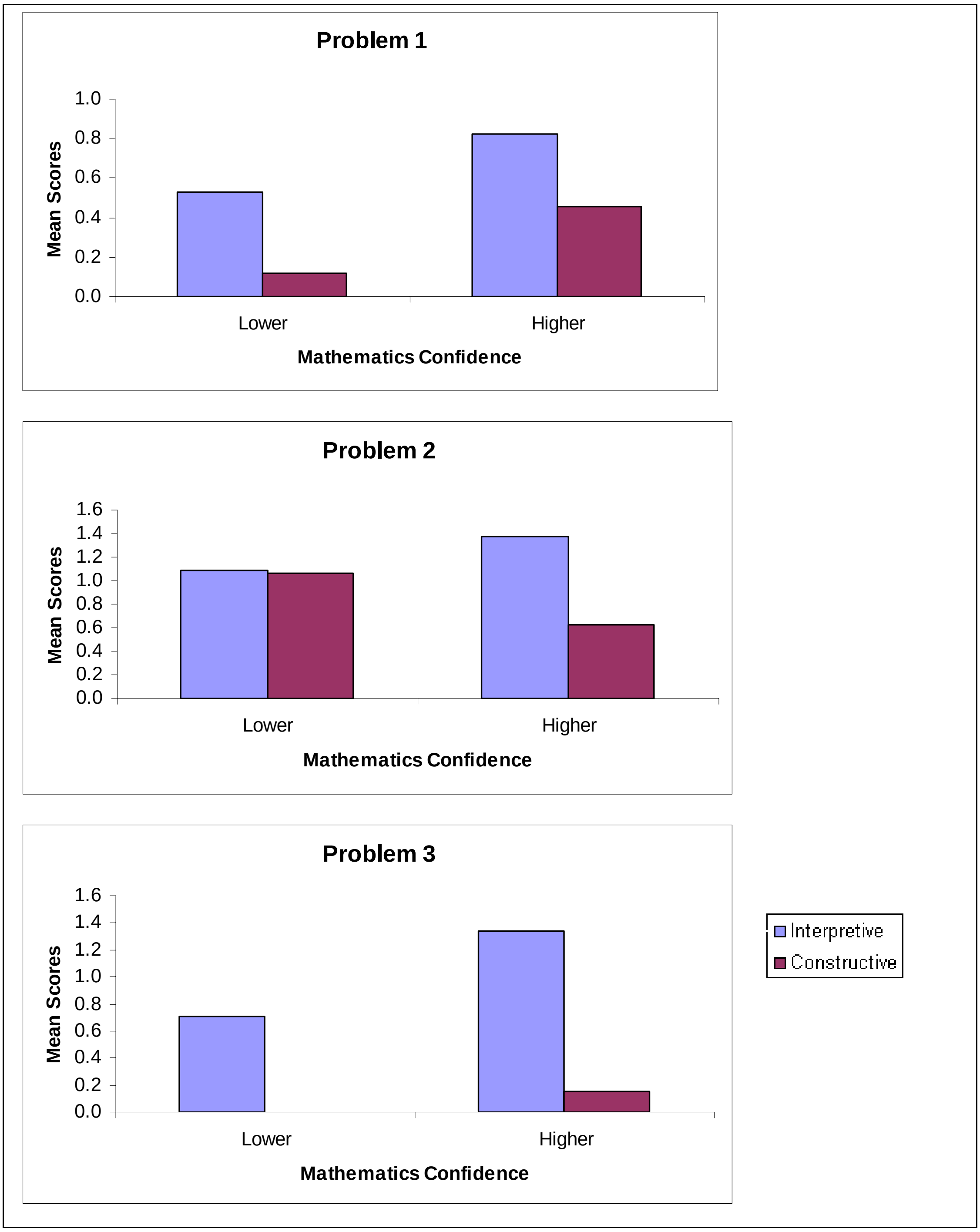


Figure 22: Mean scores for the Problems for the interpretive and constructive Tasks based on the students' Mathematics Confidence

Therefore, students' Mathematics Confidence was affecting their performance on tasks, however, only with respect to the interpretive tasks. The results for the interpretive task confirmed that there was a genuine statistical difference between the two mathematics confidence levels, as there was a large effect size ($\eta_p^2 = 0.22$). Further the performance difference in the interpretive and constructive scores confirmed the results from Galbraith and Haines (2000a). Now that performance with tasks was determined, the question to ask was whether the variation in performance was dependent on the software boxes that the students were assigned to.

5.5.2 Software Boxes and Performance

The attention is now turned to the main focus of this section, that is, whether performance varied with software boxes. In Section 5.3.2 (p.134), it was mentioned that using an ANOVA, software Boxes appeared to have no influence on performance until students' Mathematics Confidence was taken into account. From the task means for the three software boxes, students using the black-box (0.83) performed worse than those with the glass-box (1.09) and the open-box (1.01) software in the interpretive task. For the constructive task, the mean score for the black-box (0.58) was higher than the glass-box (0.30) and open-box (0.32). Figure 23 illustrates this data.

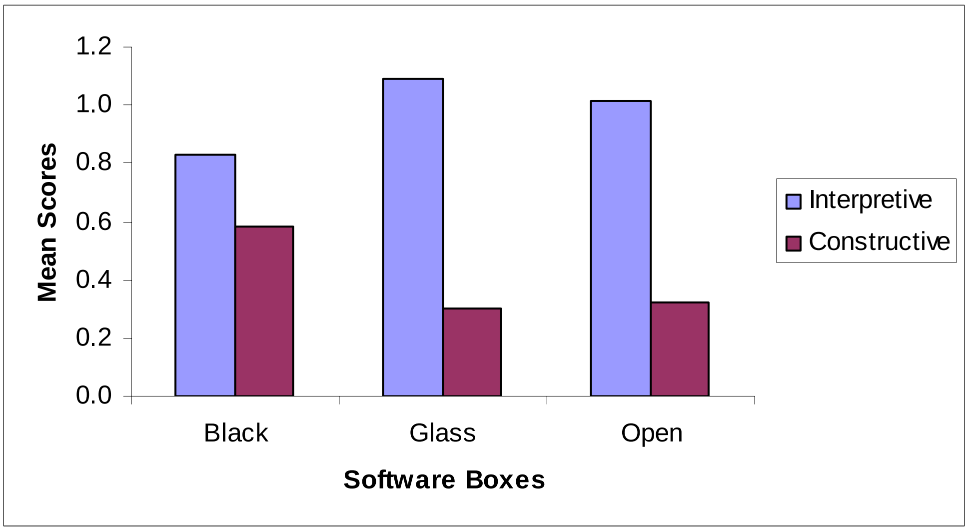


Figure 23: Mean scores for the interpretive and constructive Tasks depending on the software Box

Using Fisher's LSD, both the constructive and interpretive mean task scores were tested to determine which software Box was causing the difference. These statistical tests indicated that the scores across all three software boxes were not statistically different ($p > 0.10$) except between the black-box and the glass-box software for the interpretive task ($p < 0.08$).

Even with the marginal significance between the glass and the open-box software, the highly significant Task × Box interaction ($p < 0.01$) was still puzzling. The mean scores showed that the black-box software was under-performing and outperforming in the interpretive and constructive tasks respectively and only with the interpretive task there was any sort of significance. A graph of the mean task scores for the software boxes was then examined to determine where the significant interaction was arising. From the graph (Figure 23), a small difference between the interpretive and constructive tasks mean scores was noted for the black-box software compared to the task score differences of the glass-box and open-box software. This disparity in difference task mean scores was due to the Task × Box interaction and was confirmed using an ANOVA (see Annex 4, p.325).

From the first research question this result shows that students' performance on tasks was dependent on the software boxes. However, there is no clear answer on how the software boxes were influencing the performance. As the software boxes were used almost exclusively for the mechanical and constructive tasks (as is seen in the next section), then this result provides an indication that the black-box may be influencing the performance on constructive tasks. The higher scores in the interpretive tasks for the glass-box and open-box software may point to students engaging with these software boxes to build their conceptual knowledge but this influence is puzzling as it is not reflected in the constructive tasks which also require conceptual knowledge. Students all scored the same for mechanical tasks and hence their performance was not affected by exploration. Thus, investigating whether the students' approach for the three software boxes was different in solving the tasks and whether this influenced the scores may help in the clarification. This is now investigated quantitatively here but is discussed again in Chapter 6 from a qualitative perspective.

5.6 Exploration Approach

The three approaches of exploration, explanations and deep/surface learning are now discussed with respect to the software Boxes and Tasks. This section deals with the exploration approach and to some extent how the exploration approach affected students' performance. Section 5.7 and Section 5.8 provide results on explanations and processing levels respectively.

Students were coded as exploring (1) or not exploring (0) for the interpretive and constructive tasks depending on whether they used the software boxes or not. The mechanical task was only coded as exploring or not-exploring when students were using the software box for other purposes than just solving the mechanical task (Section 3.4.6, p.76).

Students were overwhelmingly more likely to explore for the constructive task (61%) than any of the other two tasks (see Table 25). Four of the five students who explored the mechanical tasks used the open-box software; the other student used the black-box software. This difference in exploration of tasks was due to the nature of the tasks, in that interpretive tasks were not expected to be explored although a couple of students did. Secondly, the kind of software box influenced the number of explorations in the mechanical tasks, in that students using the glass-box and black-box software were not expected to explore when doing the mechanical task as the answer was provided to them by a click of a button. However, the students using the open-box software explored the order of inputting various pivot variables to determine whether this influenced the answer (see Section 6.3.2, p.190). An explanation of how one student used the software boxes for exploring the interpretive task is presented in Section 6.4.2 (p.203).

Table 25: Number of students exploring across the three tasks

Exploration	Mechanical	Interpretive	Constructive
Not explored	33 (87%)	36 (95%)	15 (39%)
Explored	5 (13%)	2 (5%)	23 (61%)
Total	38	38	38

The values presented in Table 25 are for students exploring at least once with the software-boxes. There were three constructive tasks and some students explored all three. Hence the total number of possible explorations for the three constructive tasks per student is three. Thus for the sample of 38 students, there is a possibility of 114 explorations (38 × 3 = 114) for the constructive tasks. The same logic applies for the mechanical and interpretive tasks. Using 114 as the total number of explorations per task, the percentage of explorations by students in all of the mechanical, interpretive and constructive tasks are 5%, 2% and 33% respectively.

Again, these percentages points to students exploring more for the constructive task. As constructive tasks had the most explorations; the rest of this section deals with exploration only in the context of constructive tasks and with respect to the total number of constructive task explorations (i.e. 114).

5.6.1 Exploration and Constructive Tasks

Sequence effects were tested for exploration in the constructive task before any further analysis was completed. There was no evidence that the sequence in which problems were presented affected the number of explorations.

For the constructive task across all three problems, students explored more in Problem 2 (61%) than in any of the other problems; only seven students explored Problem 1 (18%) and eight students explored Problem 3 (21%). Taking into account mathematics confidence, six students with higher Mathematics Confidence explored

Problem 1 compared to one student with lower Mathematics Confidence. Figure 24 illustrates this data.

Although from the graph, lower Mathematics Confidence students explored more in Problem 2 (68%) than the students with higher Mathematics Confidence (53%), a chi-square test indicated that this was not significant ($\chi^2(2) = 0.99$, $p = 0.32$). Even so, this is interesting in connection with why students might have been exploring more in Problem 2. This is further discussed in Section 6.6.2 (p.226).

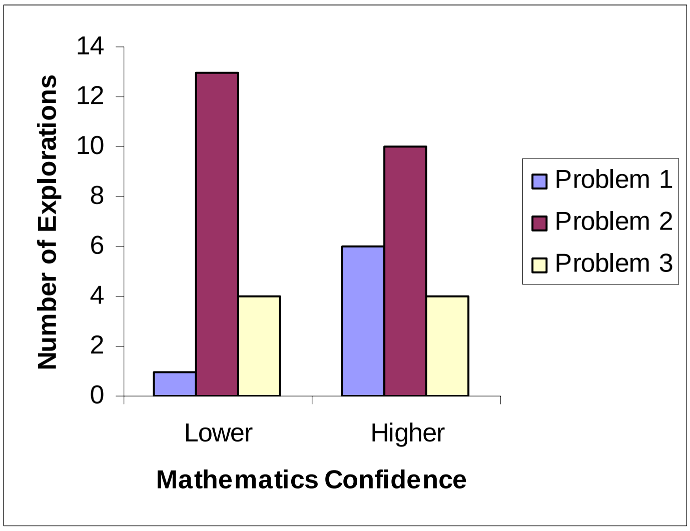


Figure 24: Number of Explorations depending on Mathematics Confidence for the constructive task in each Problem

Recall from the calculation of the possible total number of explorations for the constructive tasks across all three problems by 38 students is 114; the total number of possible explorations per software box is calculated in a similar manner. The number of possible explorations will vary as the number of students assigned to each software box is not the same. Thus, the possible total number of explorations that all students in the black-box software can make is 39 (3 constructive tasks × 13 students), for glass-box it is also 39 but for open-box software it is 36 (3 constructive tasks × 12 students). Table 26 illustrates this data for software Boxes and Mathematics Confidence.

Table 26: Number of Explorations across Mathematics Confidence and software Boxes groups for constructive tasks

Mathematics Confidence	Exploration	Software			Total
		Black	Glass	Open	
Lower	No	10	16	13	**39**
	Yes	2	11	5	**18**
	Total	12	27	18	**57**
Higher	No	12	10	15	**37**
	Yes	15	2	3	**20**
	Total	27	12	18	**57**
Total		**39**	**39**	**36**	**114**

By examining the number of explorations made across all three software Boxes, the results showed that students using the black-box software (44%) explored more than students using the glass-box (33%) and the open-box (22%) software. Whilst there was not a significant chi-square association there was a significant linear by linear association ($p < 0.01$) for the number of explorations in the three software Boxes. This shows that the number of explorations decreased linearly with the difficulty of using the software box for computing an answer.

Taking into account Mathematics Confidence, an interesting pattern emerges (see Figure 25). Students with higher Mathematics Confidence were overwhelmingly more likely to use the black-box software for exploration (56%) than lower Mathematics Confidence students in the black-box software as well as all the students using the glass-box and open-box software ($\chi^2(2) = 9.44$, $p < 0.01$). The lower Mathematics Confidence students were exploring more with the glass-box (41%) and the open-box (28%) software than with the black-box software (17%), but this was not found to be significant.

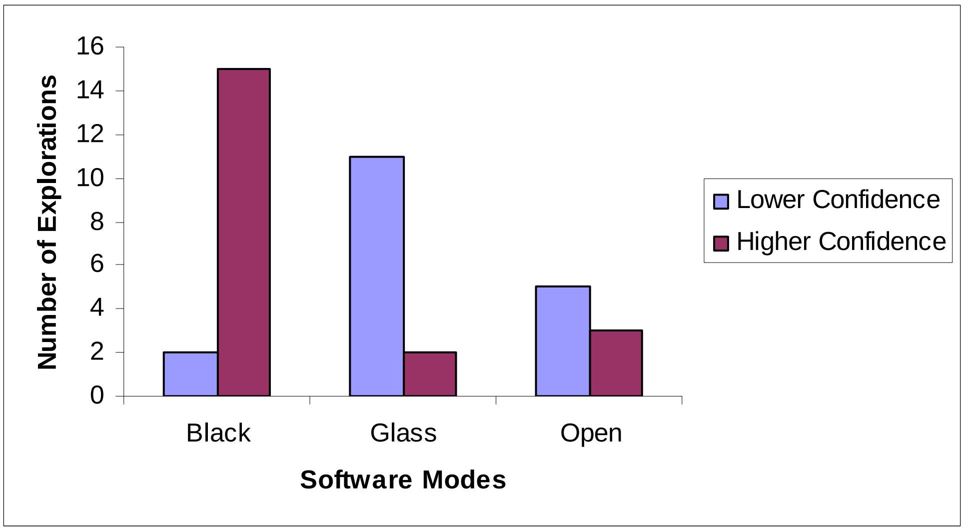


Figure 25: Number of Explorations for the constructive Task depending on Mathematics Confidence and software Box

Therefore, this section indicated that when it came to the approach of exploration, students using the open-box were more likely to explore the mechanical tasks than the students using the other two software boxes. Further, the black-box encouraged more explorations for constructive tasks and was favoured by the high mathematics confidence students. What may be more interesting is determining whether this approach had any influence on the performance.

5.6.2 Exploration and Constructive Scores

An overview of the scoring pattern of students when it came to explorations showed that if students explored the constructive task they had an 82% chance of scoring whilst those who did not explore had a 9% chance of scoring. For Problem 2, if students explored, then they all obtained a performance score. Problem 1 had slightly worse results than Problem 2, in that, if students explored, only 71% of the students were able to score. For Problem 3, the ability of students to score if they explored was low (38%). This probably points to students being unsure what to do when confronted with solving Problem 3's constructive task.

When students engaged in exploration, the percentage of times that they were able to obtain a score was 88% for the black-box software, 69% for the glass-box software and 88% for the open-box software. Thus, students who explored with the glass-box software were less likely to achieve a score than those students using the black-box and the open-box software. Perhaps as the students who explored with the glass-box software were mainly from the lower Mathematics Confidence grouping, this may indicate that they were uncertain how to solve the task with the software. This is corroborated by looking at how students with lower Mathematics Confidence were scoring based on their explorations for the glass-box software (see Table 27). These students were only able to score 64% of the time when they explored. Generally, the higher Mathematics Confidence students across all the software boxes, once they explored they were more likely to score (90%) than the students with the lower Mathematics Confidence (72%) but this was not significant.

Table 27: Number of Explorations that achieved a scored by software Boxes and Mathematics Confidence

Software	Black-Box		Glass-Box		Open-Box		Total	
Confidence	Lower	Higher	Lower	Higher	Lower	Higher	Lower	Higher
Explored	2	15	11	2	5	3	18	20
Scored	2	13	7	2	4	3	13	18
Scored (%)	100%	87%	64%	100%	80%	100%	72%	90%

Overall if students decided to explore there was a significant difference in their performance than those who did not explore. This therefore implies that using the approach of exploration may increase performance scores in constructive tasks. This section also showed that for the black-box and open-box software students, if they explored they were more likely to achieve a better performance. Students who had higher Mathematics Confidence and explored were more likely to score than those with lower Mathematics Confidence. The higher Mathematics Confidence group did however explore more with the black-box software than any other software box and the lower Mathematics Confidence students explored more with the glass-box software.

5.7 Explanation Approach

The second approach that was looked at quantitatively was explanations. Students were asked to give detailed answers for the interpretive and constructive tasks. These answers were coded into real-life and mathematical explanations based on the coding scheme presented in Section 5.3 (p.131). Since the mechanical tasks only required the students to provide a numerical solution this was not coded into these two types of explanations. Answers were only coded 13 times into both having a mathematical and real-life explanation. The reason for the two explanations coding is that conceptual knowledge was considered to be making connections with previous experiences. Therefore, the previous experiences in doing mathematical problems could probably be of two kinds. Firstly, knowledge relating to mathematics and mathematics principles and secondly knowledge from other experiences which most likely would be from social/ cultural experiences and was labelled real-life experiences.

Each student solved the three constructive and three interpretive tasks. All students, at least once, provided either a mathematical or real-life explanation when solving the tasks. Thirty-four students provided mathematical explanations and 33 students provided real-life explanations. The number of possible explanations was calculated similarly to the total number of possible explorations. There were six possible mathematical and six possible real-life explanations for both tasks combined (i.e. real-life explanations: 2 task types × 3 problems = 6 explanations). Hence for each student there was a possibility of them making 12 explanations (6 mathematical and 6 real-life). Thus for all 38 students, the total number of possible explanations was 456. The number of real-life (74) and mathematical (86) explanations made by all students was similar. This represented 35% of the total possible number of explanations.

5.7.1 A Sequence Effect

This section looks primarily at how Sequence (the sequence in which problems were answered) affected the number of explanations. By examining the total number of explanations across Sequence, it was noted that the number of explanations were affected by Sequence ($\chi^2(2) = 5.92$, $p = 0.05$). Therefore, the sequence in which problems were solved impinged on the number of explanations written by the students.

The students in Sequence 1 had the smallest percentage of explanations (27%) compared to Sequence 2 (45%) and Sequence 3 (39%). The percentages are calculated based on the total possible number of explanations per Sequence. Students in Sequence 1 started with the abstract problem (Problem 3). Perhaps as Problem 3 was mathematically focused, it encouraged students to think mathematically as well in the subsequent problems and hence discouraged real-life explanations. Therefore, the total number of real-explanations in the subsequent problems should be less (see Table 28).

Table 28: Percentage of real-life and mathematical Explanations for Problems depending on Sequence

Problem	Sequence 1: n = 12	Sequence 2: n = 14	Sequence 3: n = 12	**Total: n = 38**
Real-Life				
Problem 1	6 (25%)	16 (57%)	8 (33%)	**30 (39%)**
Problem 2	12 (50%)	13 (46%)	13 (54%)	**38 (50%)**
Problem 3	2 (8%)	2 (7%)	2 (8%)	**6 (8%)**
Total (Real-Life)	**20 (28%)**	**31 (37%)**	**23 (32%)**	**74 (32%)**
Mathematical				
Problem 1	5 (21%)	7 (25%)	4 (17%)	**16 (21%)**
Problem 2	2 (8%)	10 (36%)	14 (58%)	**26 (34%)**
Problem 3	12 (50%)	17 (61%)	15 (63%)	**44 (58%)**
Total (Maths)	**19 (26%)**	**34 (40%)**	**33 (46%)**	**86 (38%)**
Total	**39 (27%)**	**65 (45%)**	**56 (39%)**	**160 (35%)**

Note that the percentages in Table 28 are based on the total possible number of real-life or mathematical explanations in each Sequence. For example, there are two tasks (constructive and interpretive) for each problem and therefore the total number of possible real-life explanations for any problem is two (one for each task). Therefore as Sequence 1 has 12 students (*n* = 12), then the total number of possible real-life explanations made by these 12 students will be twenty-four (12 students × 2 = 24) for each problem.

However from Table 28, this is not what happened since Problem 2 actually had a lower number of mathematical explanations (8%) in Sequence 1 than in any of the other Sequences. Perhaps instead students starting with Problem 3 began to self-explain less or resorted to real-life explanations because they encountered a 'hard' problem and then was manifested in their written answers. Although Problem 3 had almost similar scores to Problem 1, it was a 'hard' problem, as the students' lack of exploration in Problem 3's constructive task suggested that they were uncertain on how to solve this task. Further, their answers as illustrated in Section 6.5.3 (p.214) and Section 6.6.3 (p.231) suggested that students had difficulty in even hazarding a guess for either the interpretive or the constructive tasks of Problem 3.

On the other hand, students who started off with an 'easy' problem (as demonstrated by their scores) such as Problem 2 (Sequence 2) were more likely to generate explanations in the subsequent problems. For example, students doing Problem 1 in Sequence 2 had the highest number of real-life explanations (57%) than in any of the other Sequences.

One conjecture was that since students performed well in Problem 2, that is, students were able to determine Problem 2's solutions quite readily; this gave them a boost of confidence. From this boost of confidence they were able to self-explain more or think of different explanations (not necessarily the right ones) and then type these various explanations. However, students starting with Problem 3 were perhaps demoralized by it being ‘hard’ or abstract and did not feel like self-explaining any more. In Sequence 2, Problem 1 was done after the ‘hard’ Problem 3, and it may seem that whilst students in Sequence 2 generated more explanations, following the ‘hard’ problem, the students decided to generate more real-life explanations for Problem 1. This may be an indication of the efficacy-performance spiral (Gist and Mitchell, 1992; Lindsley, Brass and Thomas, 1995). The efficacy-performance spiral suggests that when students are not able to solve tasks correctly (performance), they lose confidence. However by losing confidence, they affect their approach to solving tasks in future since they are no longer keen to investigate different task-solving strategies (such as making more explanations). Hence their performance drops and the vicious cycle continues.

Also, there are studies that indicate that students perform better when tasks are placed from easy to hard (e.g. Plake, Ansorge, Parker and Lowry, 1982; Towle and Merrill, 1975). A performance difference was not seen between Sequences in this study probably because there were only three problems and students were randomly assigned to each Sequence. Possibly this known occurrence is manifested in this study through the number of explanations. Perhaps if a larger sample of students, more problems and more permutations of the problem sequences were used then the effect of problem sequence on explanations and performance can be observed.

Thus to sum up, there was a sequence effect affecting the number of explanations, with students who started off with Problem 3 (i.e. those following Sequence 1) being less likely to generate any explanations. The suggested reason for this effect was that students were less likely to self-explain and type these explanations following this ‘hard’ problem probably because of an efficacy-performance spiral effect.

5.7.2 Explanations: Problems, Tasks and Boxes

Whilst a Sequence effect is found, when looking at the number of explanations across Problems, Tasks and software Boxes, one should note that students were randomly assigned to Sequence. Each software Box had a similar number of students in each Sequence which meant that any effect from Sequence will be evened out across Problems, Tasks and Software Boxes.

Amongst the three problems, students made slightly more explanations for Problem 2 (64 explanations) than Problem 1 (46 explanations) and Problem 3 (50 explanations). The abstract problem, Problem 3, had a very low percentage of real-life explanations (8%) compared to the two application problems, Problem 1 (39%) and Problem 2 (50%). However, Problem 3 had a high percentage of mathematical explanations (58%) compared to Problem 1 (21%) and Problem 2 (34%). Percentages are calculated based on the possible number of real-life or mathematical explanations for each problem (i.e. total number of possible real-life explanations per problem: 38 students × 2 tasks per problem × 1 real-life explanation = 76 real-life explanations). Figure 26 presents these results.

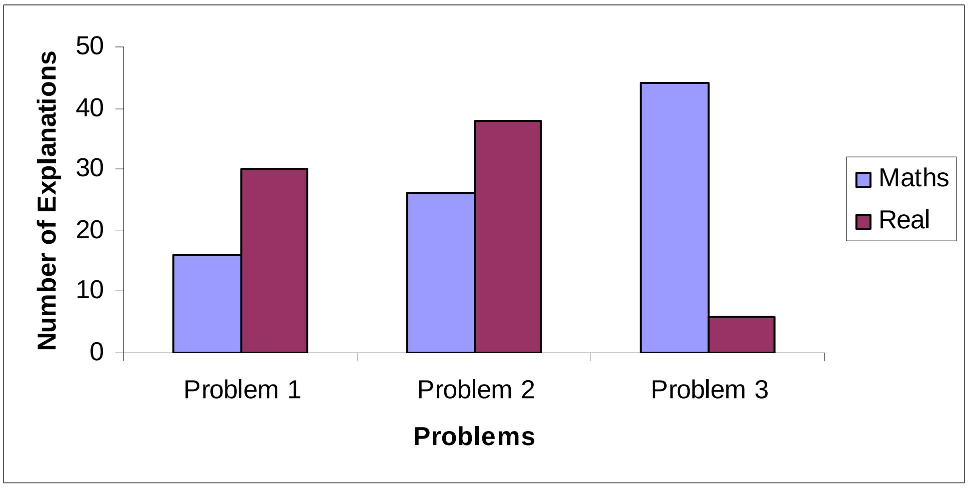


Figure 26: Number of mathematical and real-life Explanations for each Problem across all students

These results suggest that if there is a real-life application task, students were more likely to rely on making real-life type explanations and to a lesser extent mathematical explanations to support their answer. However if the task is abstract then students probably did not see the relation to a real-life situation and rely mostly on mathematical explanations (Section 6.4.2, p.203). With the application problems, students probably brought in real-life knowledge to help understand the task (Boaler, 1993), whether this hampered their performance or understanding is ascertained in the next section and Chapter 6.

Overall there were more explanations for constructive tasks (93) than interpretive tasks (67). Constructive tasks had more than 1.5 times the number of mathematical explanations (47%) than the interpretive task (28%). The bracketed percentages are based on the possible number of mathematical explanations per task (i.e. 114). Students made almost equal numbers of real-life explanations between both tasks (see Table 29).

Table 29: Number of real-life and mathematical Explanations for Problem and Task

Tasks	Interpretive		Constructive	
Explanations	Maths	Real	Maths	Real
Problem 1	5 (13%)	5 (13%)	11 (29%)	25 (66%)
Problem 2	8 (21%)	26 (68%)	18 (47%)	12 (32%)
Problem 3	19 (50%)	4 (11%)	25 (66%)	2 (5%)
All	32 (28%)	35 (31%)	54 (47%)	39 (34%)

Given that students made more mathematical explanations in Problem 3, this was reflected in the high number of mathematical explanations in the interpretive (50%) and constructive tasks (66%) and the low number of real-life explanations in both of these tasks for this problem, as seen in Table 29. The percentages are based on the total possible number of explanations, per task per problem which is 38. Therefore the number of explanations for the task per problem also represents the number of students making that type of explanation. Students gave more real-life explanations for Problem 2's interpretive task (68%) than any other interpretive task whilst Problem 1's constructive task attracted the most real-life explanations (66%) amongst all the constructive tasks. The real-life explanations were the predominant explanation type for these two tasks (i.e. Problem 1's constructive task and Problem 2's interpretive task). Problem 1's interpretive tasks and Problem 2's constructive task had similar number of mathematical and real-life explanations.

Now whilst all of this is interesting, what is important for this thesis is to know whether the software Boxes influenced the number and type of explanations. Using a chi-square test, only marginal significance ($\chi^2(2) = 5.19$, $p = 0.07$) was found such that the software Boxes were associated with the number of real-life explanations (see Figure 27). Students using the glass-box software (41%) were more likely to have a higher number of real-life explanations than those students on the black-box (32%) or open-box (24%) software.

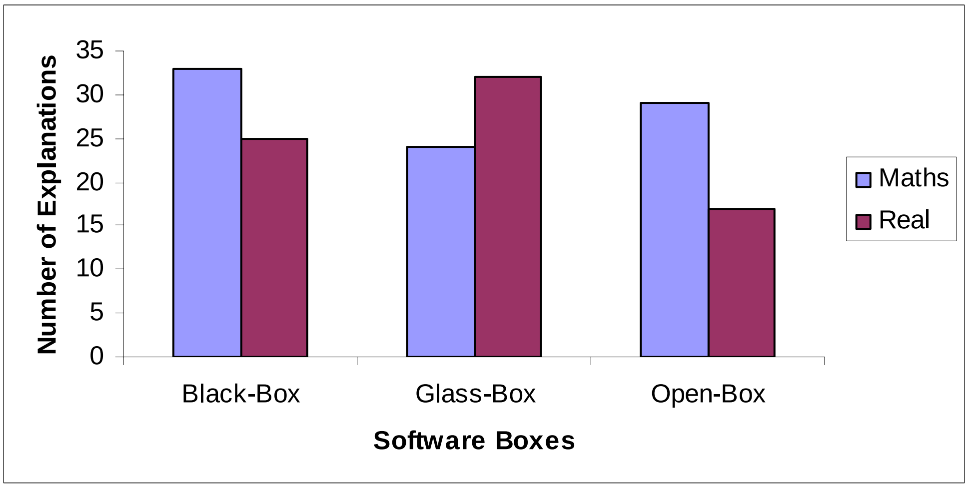


Figure 27: Number of real-life and mathematical Explanations for the software Boxes

This association of real-life Explanations and software Boxes was mostly due to the students with higher Mathematics Confidence ($\chi^2(2) = 7.65$, $p = 0.02$). Higher mathematics confidence students using the glass-box software provided real-life explanations for half of their tasks (constructive and interpretive combined), whilst higher mathematics confidence students in the open-box software barely gave any real-life explanations (17%). Higher mathematics confidence students using the black-box software provided real-life explanations for 30% of their tasks.

Perhaps the open-box software because of its mathematical steps made students feel that any explanations should be more mathematical in nature and hence they reduced their number of real-life explanations. The reason why the higher mathematics confidence students using the glass-box were providing more real-life explanations is uncertain. However, as the higher mathematics confidence students did not explore with the glass-box greatly, they probably tried to explain their given answers by providing real-life explanations. The ratio of mathematical to real-life explanations for all students was found to be the lowest in the glass-box software (0.75) and the highest in the open-box software (1.71). Black-box software had a ratio of 1.32.

Thus to sum up this section, students solving the constructive task gave the most number of explanations. The constructive task asked students to explain what will happen if a value changes (such as profit or number of products) and this was followed with why the change (or no change) occurred. This probably gave students a wider berth to use a range of explanations. Mathematical explanations seemed to be more popular for Problem 3, for both its interpretive and constructive tasks. The number of real-life explanations was popular in one interpretive task (Problem 2) and one constructive task (Problem 1) which were both related to application problems (Problems 1 and 2). Finally, higher mathematics confidence students were more likely to provide real-life explanations if they were using the glass-box software.

5.7.3 Explanations and Performance

This section investigates student's performance scores based on their explanations. Of the total number of tasks answered by all students (228 i.e. 38 students

× 3 problems × 2 tasks), only for 133 of these tasks was a score obtained. Hence, the total possible number of explanations that can be made for these scored tasks will be 266 (i.e. 133 tasks × 2 explanations). Students had the same percentage of explanations (35%) for the scored tasks (94 explanations) and non-scored tasks (66 explanations) based on their respective possible number of explanations. This shows that even if students were unable to obtain the correct answer, they still provided an explanation to justify what they wrote.

Students, who were unable to score in Problem 1, provided more real-life explanations (60%) than those students who did score (17%), ($\chi^2(1) = 14.89$, $p < 0.01$). The few students, who were unable to score in Problem 2, all provided real-life explanations (see Table 30). Regardless of how students performed on Problem 3, they provided substantially more mathematical explanations to explain their answer. It appears that for the application problems that students' lack of performance is associated with real-life explanations.

Table 30: Number of real-life and mathematical Explanations by Problem for scored and non-scored Tasks

	Scored Tasks			Non-Scored Tasks		
	Real-Life	Maths	**Tasks No.**	Real-Life	Maths	**Tasks No.**
Problem 1	6 (17%)	9 (25%)	**36**	24 (60%)	7 (18%)	**40**
Problem 2	31 (49%)	26 (41%)	**63**	7 (54%)	0 (0%)	**13**
Problem 3	4 (13%)	18 (53%)	**34**	2 (5%)	26 (62%)	**42**
Total	**41 (31%)**	**53 (40%)**	**133**	**33 (35%)**	**33 (35%)**	**95**

Looking at the interpretive task, there were 95 scored interpretive tasks and 19 non-scored interpretive tasks. Students had a slightly higher percentage of explanations for scored (61%) and non-scored (47%) for this task but it was not significant. For the constructive task, out of the 38 scored tasks, 26 of these had mathematical explanations (68%). For the non-scored constructive tasks, only 28 of the 76 tasks (37%) had any mathematical explanations. A chi-square test ($\chi^2(1) = 10.13$, $p < 0.01$) indicated that there was a positive association between the number of mathematical explanations and the performance scores for the constructive task. This association seemed primarily due to the high percentage of mathematical explanations in the constructive tasks of Problems 1 and 2 between the scoring and non-scoring tasks (see Table 31).

Table 31: Percentage of real-life and mathematical Explanations for scoring and non-scoring constructive Tasks

	Scoring Constructive Tasks			Non-Scoring Constructive Tasks		
	Real-Life	Maths	**Tasks No.**	Real-Life	Maths	**Tasks No.**
Problem 1	4 (44%)	6 (67%)	**9**	21 (72%)	5 (17%)	**29**
Problem 2	5 (19%)	18 (69%)	**26**	7 (58%)	0 (0%)	**12**
Problem 3	1 (33%)	2 (67%)	**3**	1 (3%)	23 (66%)	**35**
Total	**10 (26%)**	**26 (68%)**	**38**	**29 (38%)**	**28 (37%)**	**76**

The effects of Mathematics Confidence and software Boxes were also investigated for explanations across the scoring and non-scoring tasks and there were no significant associations.

5.8 Processing Levels

The two approaches of exploration and explanations discussed thus far were with respect to each task type. However, for the third approach, the deep/surface processing levels, this was not possible, since the processing levels were determined for the whole session. The processing levels were measured through the SOMUL Approaches to Study Inventory (ASI) (Section 3.4.5, p.75). Thus, determining whether the deep/surface processing levels differed for task was not possible. However it was possible to determine whether this differed for the software Boxes and Mathematics Confidence (Section 5.3.1, p.132).

5.8.1 Processing Levels and the Main Variables

From the ASI, the students' scores on both the surface (out of 20) and deep (out of 30) scales were obtained. Only 37 of the 38 students' ASI scores were obtained as one was lost due to technical problems. An ANOVA was used to determine whether there was a significant difference for the processing levels depending on software Boxes, Sequence or Mathematics Confidence (see Table 32).

Table 32: ANOVA for Deep and Surface scores depending on Box, Mathematics Confidence and Sequence

Source	SS	df	MS	F	P	η_p^2
Surface						
MathConfRec	6.10	1	6.10	0.83	0.37	0.04
Sequence	100.72	2	50.36	6.85	0.01	0.40
Box	2.06	2	1.03	0.14	0.87	0.01
MathConfRec × Sequence	9.83	2	4.92	0.67	0.52	0.06
MathConfRec × Box	32.79	2	16.39	2.23	0.13	0.18
Sequence × Box	60.51	4	15.13	2.06	0.12	0.28
MathConfRec×Sequence×Box	9.63	2	4.82	0.66	0.53	0.06
Error	154.50	21	7.36			
Deep						
MathConf Rec	10.49	1	10.49	0.68	0.42	0.03
Sequence	5.67	2	2.83	0.18	0.83	0.02
Box	7.20	2	3.60	0.23	0.79	0.02
MathConfRec × Sequence	3.81	2	1.91	0.12	0.89	0.01
MathConfRec × Box	25.52	2	12.76	0.83	0.45	0.07
Sequence × Box	10.48	4	2.62	0.17	0.95	0.03
MathConfRec×Sequence×Box	4.92	2	2.46	0.16	0.85	0.02
Error	324.92	21	15.47			

Firstly, students scored low on the surface scale (9.2 out of 20) compared with that of the deep scale (24.1 out of 30). This meant that students used mostly a deep processing level and less of a surface processing level during the study. Students generally had high Mathematics Confidence, that is, 34 of the 38 students assessed themselves as having a Mathematics Confidence of at least 5. Therefore, as deep processing level is associated with mathematics confidence then this was probably the reason why the students had such high deep scores and low surface scores.

The only significant effect from this ANOVA was that students' surface scores were significant for Sequence ($F(2,21) = 6.85$, $p < 0.01$, $\eta_p^2 = 0.40$). Students who had been assigned to Sequence 1 (i.e., they started with Problem 3 – the abstract problem) were likely to have a lower surface processing level score (6.52) than those who were assigned to Sequence 2 (10.92). This was noteworthy since it followed a similar pattern to the number of explanations for Sequence (see Figure 28, p.163).

Students with a low number of explanations were expected to have a more surface processing level but perhaps the quality of explanations exceeded the quantity, because the scores obtained by students in all Sequences were similar (no sequence effect for scores). Also the number of explanations was not an indicator that students were not doing well, but rather the type of explanations (i.e. mathematical explanations).

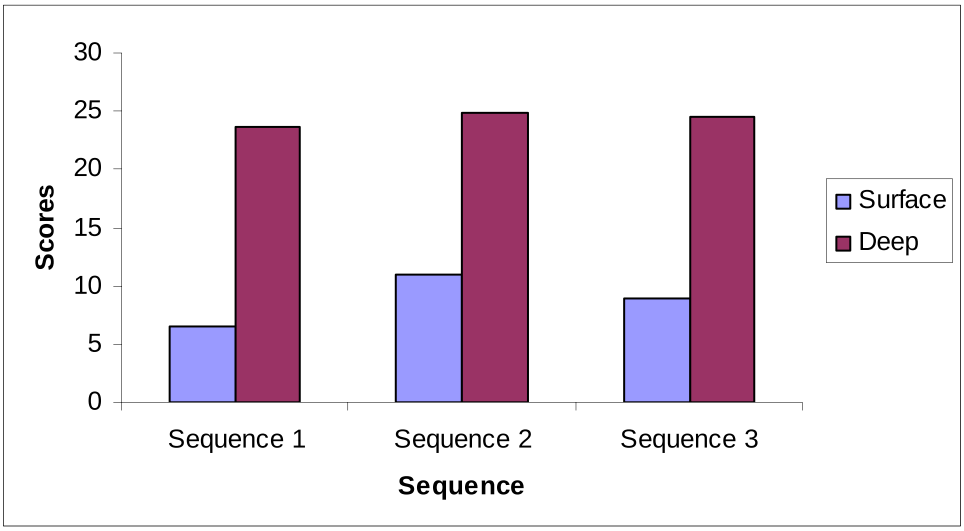


Figure 28: Deep and Surface Scores for students depending on Sequence

However, the argument made in the Section 5.7.1 (p.153) was that students assigned to Sequence 3 were probably demoralised and hence they self-explained less. However, the ASI scores showed that students who started in Sequence 1 were more likely to have a lower surface score. Further, the ANOVA for Sequence and Surface scores indicated that there was a significant difference between the Surface scores obtained for Sequence 1 and Sequence 2. This may mean that the experience of solving the tasks in Sequence 2 led to higher surface scores.

It was possible that instead of Sequence 1 actually demoralizing students it kept them on track with the explanations that were required to answer the subsequent problems. Whilst the students in Sequence 2, because they were thinking of a wider range of possibilities, that is, both mathematical and real-life explanations, they got more confused and probably made more explanations but not necessarily deep and connected explanations. Hence, the conjecture that in Sequence 1 they were making more minimalist explanations since they knew which explanations would possibly aid in answering the task. Whilst the surface scores were different, the deep scores were the same for all groups of students and the question arises whether the deep processing level influenced the performance as was expected.

5.8.2 Processing Levels and Performance

To ascertain whether the deep/surface processing levels were related to the performance of students for the Tasks and software Boxes, correlations were performed between the deep and surface scores and performance scores which showed that there were no significant relationships (Table 33).

Students with lower mathematics confidence scored 9.5 and 25.0 for the surface and deep scales respectively. Those students with a higher mathematics confidence scored 8.8 and 23.2 for the same scales respectively. From Chapter 2, the literature indicated that the deep processing level should be aligned with the performance scores and from Chapter 4 that the processing level scores should be related to the mathematics confidence. Based on these values, there is no disparity perhaps because the SOMUL ASI with only its 10 items was not able to discern sufficiently between the two groups of mathematics confidence considering that students had a relatively high mathematics confidence in this study.

Table 33: Correlation coefficients between the processing levels and task scores

	Surface Score (n = 37)	Deep Score (n = 37)
Mathematics Confidence	0.01	-0.28
Interpretive Task Score	-0.06	0.05
Constructive Task Score	-0.13	-0.29
Total Score	-0.12	-0.15

5.9 Discussion

This section provides a discussion on all the results found in this chapter and tries to align these results with the identified research questions.

5.9.1 Performance due to Tasks and Boxes

The performance scores were dependent on the problem (Section 5.5.2, p.144). Students performed well in Problem 2, which was an application problem. The scores for the other application problem (Problem 1) and abstract problem (Problem 3) were lower than that of Problem 2. Students may perform better in the application problems; however, the experimental design did not allow an exhaustive look at the abstract and application problems. Further, it was found that whilst mathematics confidence did not influence how students performed in Problem 2, it did influence the performance in Problems 1 and 3, where students with higher mathematics confidence were more likely to do better. Therefore, students with higher mathematics confidence should mostly perform better than lower mathematics confidence students as expected but there may be problem types where mathematics confidence may not always influence performance.

Performance was also found to be associated with tasks (Section 5.5.2, p.144). The scores obtained for the interpretive and constructive tasks corroborated the prediction and results of Galbraith and Haines (2000a) that students performed better in interpretive tasks than constructive tasks. The study by Galbraith and Haines was based on students solving these two task types with pen and paper. In the current study where there is a technology-enabled environment, students performed better in the interpretive tasks than in the constructive tasks. Although students had access to technology to solve the procedural part of the constructive task, this did not provide any advantage to the students, possibly as knowing what to do with the software box rather than being able to use the software box played an important role in the performance of the constructive task.

A comparison of scores obtained by Galbraith and Haines for their undergraduate students solving tasks in polynomial algebra showed that their students scored 37% less in the constructive task than in the interpretive task. In the present study, the students performed even worse as they scored 60% less in the constructive task than the interpretive task. This probably is because the two studies were in two different mathematical domains or possibly because Galbraith and Haines used more tasks (6 each).

The scores obtained for the interpretive and constructive tasks also varied with Problem, where students performed better in the interpretive tasks for Problems 2 and 3. The suggested reason for this was because the answers for these two interpretive tasks had to be deduced from previously calculated values and these required interpreting or reading off the variables carefully. For the constructive tasks, students all performed significantly different on each problem, performing best on Problem 2's constructive tasks and the worst in Problem 3's constructive task. This may point to an effect of problem type for the constructive task, with the students performing better on the applications problems than the abstract problem.

Task types were also found to be affected by mathematics confidence with students who had higher mathematics confidence performing better in the interpretive tasks, whilst performing equally well in the constructive task regardless of mathematics confidence. This probably meant that since interpretive tasks required conceptual understanding that these students with higher mathematics confidence compared to lower mathematics confidence students were more likely to make connections between the answers ascertained from the mechanical task and then deduce what they meant, such as in the cases of Problems 2 and 3.

It was interesting that the students regardless of mathematics confidence performed the same in the constructive task, as this required finding relationships between procedural and conceptual knowledge and then applying them together. It was expected that the higher mathematics confidence students were probably more poised to do these tasks since they had a higher likelihood of undertaking a deep processing level and connecting their procedural and conceptual knowledge.

Now, looking at the first research question, that is:

Does the students' performance in solving the three task types depend upon the software box they have access to?

What has been noted was that performance was dependent on task type but this was already known from Galbraith and Haines. Now, was performance on the tasks influenced by the software boxes? A cautionary yes is put forward. Performance on either the interpretive or the constructive tasks was not significantly dependent on the software boxes, although the difference in performance scores between the interpretive and constructive scores was (i.e. the interpretive score minus the constructive score) dependent on the software boxes (Section 5.5.2, p.144). In particular, students using the black-box software had the smallest disparity in their scores compared to the students using the glass and open-box software. Further, marginal significance showed that students using the glass-box software were doing better in the interpretive tasks than those students in the black-box. Although there was no statistical significance, graphical trends indicated that students using the black-box were performing best in the constructive task. This suggested that for the constructive tasks, students using the black-box software were more likely to grasp the conceptual-procedural knowledge connection required for solving constructive tasks and perhaps the black-box software influenced the ease in which the students were able to make this connection.

5.9.2 Exploration Approach for Tasks and Boxes

The exploration approach for only the constructive tasks was looked at in detail since for the mechanical and interpretive tasks very few students explored these since they were probably not tasks that encouraged exploration. Students who explored the mechanical tasks were more likely to be students using the open-box software since it allowed the exploration of different processes such as in the choosing different pivot variables.

Therefore, using only the constructive task to investigate explorations, it was found that students explored more with the black-box (44%) than the glass-box (33%) and the open-box software (22%) (Section 5.6.1, p.147). Students also explored more for Problem 2's constructive task and least for Problems 1 and 3's constructive task. The number of explorations was found to be dependent on students' mathematics confidence. Students who had higher mathematics confidence were more likely to explore Problem 1 than the lower mathematics confidence students but all students regardless of mathematics confidence explored similar amounts for Problem 2 and 3.

Now, the research question to be answered:

Does the students' exploration approach when solving the three task types depend upon the software box they have access to?

Again, the answer to this question is a cautionary yes as it is with only with respect to the constructive tasks. The exploration of tasks was dependent on the software boxes with the black-box software having the most exploration. The students with higher mathematics confidence were more likely to explore using the black-box software. Also, the students with the black-box software scored higher in the constructive tasks. The link between this approach and performance is now looked at.

How is the students' exploration approach when solving the three task types associated with their performance? And does this vary with the software box they have access to?

First of all, there is evidence to show that the exploration approach affects the performance. However the influence of exploration was dependent on the problem and the students' mathematics confidence. In general, if students decided to explore they scored higher than students who did not explore (Section 5.6.2, p.150). However the greatest disparity was dependent on whether students explored for Problem 2's constructive task, because if students explored this task, there was a good chance of them scoring. This was quite different for students exploring Problem 3, since whether the students explored or did not explore, their scores were similar.

Now with respect to whether the software boxes had any influence, it appeared that students who had higher mathematics confidence were marginally more likely to have a higher score if they explored with the black-box software than all the other students (low mathematics confidence or using glass or open-box software). Therefore, the software box that students had access to as well as students recognising a conceptual-procedural link influenced the performance when using this approach.

5.9.3 Explanations Approach for Tasks and Boxes

The number of explanations varied with Sequence. The Sequence effect was dependent on whether students started with Problem 3 (abstract). If students started with this problem they were less likely to generate explanations (Section 5.7.1, p.153). Whilst the number of explanations may have decreased, the quality of explanations leading to higher performance scores probably was not affected, as was demonstrated by the number of explanations having no influence on the performance scores (Section 5.7.3, p.159).

The number of explanations also varied with Problems and Tasks (Section 5.7.2, p.156). Problem 2 generally had more explanations possibly because it was ‘easy’ (as demonstrated by their scores). Thus, students probably felt better to posit explanations whether they be real-life or mathematical. Students were more likely to give more explanations for the constructive tasks and less for the interpretive tasks.

Further it was found whilst quantity of explanations did not affect the performance, the type of explanations did (Section 5.7.3, p.159). Students who used mathematical explanations were more likely to score higher, particularly in the constructive tasks. The data also suggested that higher mathematics confidence students using the glass-box software were more likely to generate more real-life explanations. The possible reason for this is still unclear. The higher mathematics confidence students using the open-box software provided more mathematical explanations and less real-life explanations (a ratio of 2.67). Perhaps for the students using the glass-box software, just seeing the steps did not encourage them to engage with the software in a mathematically-minded manner unlike the students in the open-box software and hence relied mainly on real-life explanations to help support their answers.

Now to answer the research question:

Does the students' explanation approach when solving the three task types depend upon the software box they have access to?

The answer is yes to some extent. The type of explanations did vary across software boxes but was dependent on mathematics confidence (Section 5.7.2, p.156). Higher mathematics confidence students using the glass-box software provided more real-life explanations than the higher mathematics confidence students using the open-box software, where the latter students preferred less real-life explanations but favoured more mathematical explanations.

Further the type of explanations also differed across tasks. For the interpretive task, students had a preference for more real-life explanations than mathematical explanations. This suggested that perhaps when students approached the solving of tasks they will try and make sense of the task from their knowledge experience, either using their mathematical or real-life knowledge. Further, the type of problem also may influence which explanations that they will use, in particular, if they had an abstract problem they will use a mathematical explanation, however, application problems will most likely cause students to use real-life explanations. Thus, when students are solving an application problem the students will try and make sense of this from a real world experience and from a mathematical perspective equally.

How is the students' explanation approach when solving the three task types associated with their performance? And does this vary with the software box they have access to?

The students’ explanation approach was shown to influence performance. The total number of explanations did not influence the performance of the students but the number of real-life or mathematical explanations did (Section 5.7.3, p.159). Generally there was a positive association between mathematical explanations and their performance on constructive tasks for two out of three problems. The number of explanations was not influenced by software box and mathematics confidence and hence students' performance with regards to explanation was not influenced by software box.

5.9.4 Processing Levels Approach with Tasks and Boxes

For the final approach, the processing levels were investigated. As deep and surface scores were measured over the study period rather than on a task by task basis, the students' processing levels were analysed within this context and thus only their influence on students' performance was determined quantitatively. The processing levels approach is examined at the task level qualitatively in Chapter 6.

How is students' processing levels approach when solving the three task types associated with their performance? And does this vary with the software box they have access to?

Quantitatively, no apparent link was found between performance and the deep and surface scores or that the deep or surface scores varied with the software boxes, but this issue is also investigated qualitatively in Chapter 6.

5.10 Concluding Remarks

This chapter presented the results from the quantitative investigation of the links between software boxes, tasks, performance and the three approaches (Sections 5.5 to 5.8) and discussed how these results answered the research questions (Section 5.9, p.165).

Software boxes were found to influence performance (Section 5.5, p.139). Further, students having access to the black-box software were more likely to explore (44%) than the students using the other two software boxes (33% glass-box software and 22% open-box software) (Section 5.6.1, p.147). Open-box software promoted more mathematical explanations whilst the glass-box software promoted more real-life explanations (Section 5.7.2, p.156). Mathematical explanations were associated with better performance but only on constructive tasks. There was no apparent influence of software Boxes on the association of mathematical explanations and performance. No quantitative relationship was found between the deep or surface processing levels and the software boxes or performance (Section 5.8, p.161).

The detailed quantitative findings are summarised in Appendix 7 (p.315) for performance scores (Annex 5, p.326), frequency of exploration (Annex 6, p.328), exploration and performance scores (Annex 7, p.329), frequency of explanations (Annex 8, p.330), explanations and performance scores (Annex 9, p.332), and processing levels (Annex 10, p.332). Each quantitative finding is numbered and would be referred to by its finding number in Chapter 6.

In this chapter it was noted that the influence of students' deep and surface processing levels and quantity of explanations could not conclusively be determined as to why there were differences in tasks and software boxes. The literature review in Chapter 2 suggested that students' processing levels and quality of explanations should affect how students solved the tasks. Chapter 5 dealt with the quantitative data in particular students' performance scores. Thus, it is possible that even if students produced the wrong answer that the students' processing levels and explanations could still differ when arriving at their answers. Thus, in Chapter 6, the explanations and the students' processing levels are discussed qualitatively for the software boxes and tasks. The explanations and processing levels are considered qualitatively for each task by first understanding the findings from Chapter 5 based on the students' performance scores and mathematics confidence and further in the case of the constructive tasks, their exploration habits. Approximately 11 main Findings (Findings 2 to 5, 7 to 8, 11 to 15) are drawn upon to help further this examination.

Chapter 6. Tasks, Boxes and the Approaches: Qualitative Analysis

"The mind uses its faculty for creativity only when experience forces it to do so. "
- Jules Henri Poincare

6.1 Introduction

This chapter deals with Research Question 3 and the overarching research question. These questions seek to know how performance is influenced by the approaches and how this influence varies across the software boxes. In Chapter 5, the quantitative data showed that students' performance on task types were dependent to some extent on explanation types, students' mathematics confidence and explorations. Further, the performance was also dependent on the software box that the students had access to. This chapter tries to describe how or why there is a relationship between performance and approaches by looking at how the students solved and answered the tasks, using the collected think-aloud data, the typewritten answers and the screen capture videos of the student and the software boxes.

This chapter begins by reflecting on the qualitative data collected (Section 6.2, p.175). The chapter is then structured into three main sections representing each of the task types: that is, mechanical (Section 6.3, p.181), interpretive (Sections 6.4 and 6.5) and constructive (Section 6.6, p.220). Each section then discusses how performance was affected by the approaches as well as mathematics confidence. This is then followed up by looking at how the different software boxes impacted on these relationships. When discussing the interpretive and constructive tasks, the analytical unit used for comparison is the problem. The analytical framework presented in Figure 3 (p.48) is used as a lens to investigate the relationships. The updated framework includes the finding from Supporting Study 1 that mathematics confidence is related to mathematical processing levels. The chapter finishes with how the results presented both here and in

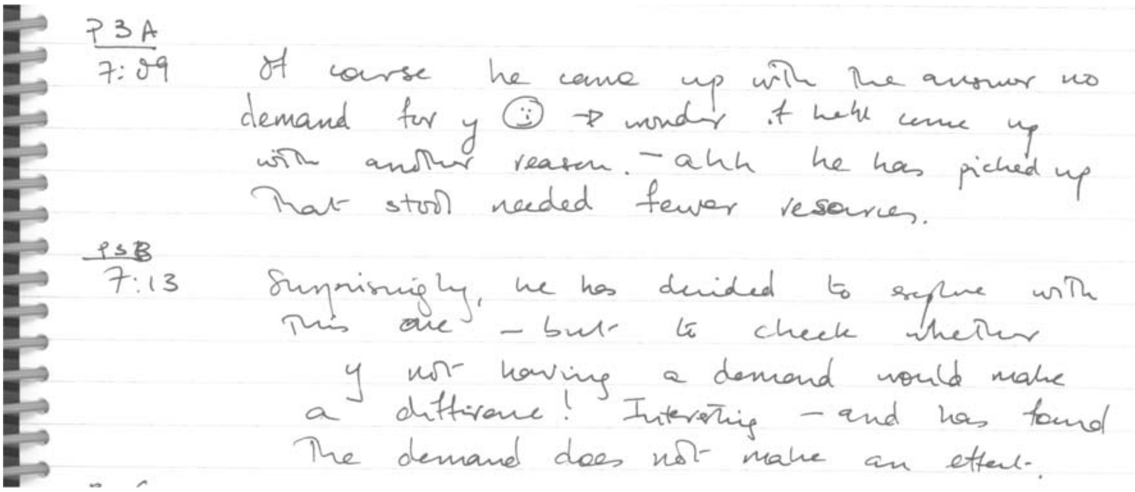
Chapter 5 help to answer the research questions (Section 6.7, p.237) and then concluding remarks are made (Section 6.8, p.242).

6.2 Think-Aloud and Typewritten Data

The tasks on the post-test were marked on a scheme (Appendix 6, p.313). The marking scheme provided the expected answers but any variation or answers that were near to these were accepted, particularly when students answered the tasks from a mathematical aspect.

During the post-test, which lasted approximately one hour, the students were asked to think-aloud (Ericsson and Simon, 1984) as they worked through the tasks. Students were prompted to think-aloud by using phrases such as “Keep talking”, “What are you thinking?” and “What are you doing now?” It was necessary to use the latter two phrases when students were not forthcoming or were engrossed in working with the software or watching the screen. Students were not prompted to talk whilst inputting data or typing in their answers. As indicated in Section 5.2 (p.125), the audio session was either missing or corrupted in the case of four participants (Participants 5, 13, 8 and 26). The typewritten answers for the tasks by all 38 students came to over 11,210 words: that is, approximately 295 words per student. In addition, over 40 pages of observation notes by the researcher were taken (see Figure 29).

To support this data, 8 students’ sessions were transcribed. This is similar to the number of transcripts used by Chi *et al.* (1989) in her self-explanations study where 8 students’ transcripts were also compared. The eight students in this study were chosen to be representative of their assigned software box, their mathematics confidence and the sequence they solved the problems. The students were selected based on at least one student representing each software box in the higher and lower mathematics confidence groups. Also, for each mathematics confidence grouping, at least one participant had to represent one of the sequences.


P3A
7:09 Of course he came up with the answer no demand for y ☺ I wonder if he'll come up with another reason. – ahh he has picked up that stool needed fewer resources.

P3B
7:13 Surprisingly, he has decided to explore with this one – but to check whether y not having a demand would make a difference! Interesting – and has found the demand does not make an effect.

Figure 29: Example of observational notes for Participant 13 whilst solving Problem 2 (Question 3)

Further, students were chosen based on their performance scores and how vocal the students were, that is, the frequency of speech during the think-aloud session. The students' frequency of speech allowed the researcher to investigate more fully the self-explanation process and hence students who were more verbose were considered for selection. Also, students with average to high scores were more likely to be considered for selection within the groupings because these were the students who provided more written or think-aloud explanations as some students often wrote "I don't know" without providing any think-aloud explanations. These students with minimal think-aloud or written explanations were still used for illustrative purposes when discussing qualitative data in this chapter with respect to their written answers. As mathematics confidence was found to influence performance scores, a high and low performance scorer from the higher (Participants 9 and 39 respectively) and lower mathematics confidence groupings (Participants 33 and 6 respectively) were also chosen.

The participants whose sessions were transcribed are presented in Table 35. The term 'Part.' in the table represents participant. N/P in the table indicated that there was no participant representing that combination and 'Missing audio' indicated that there was one participant who fit the profile but the audio was missing. The blanks represent combinations that were not transcribed.

Table 34: Participants whose session was transcribed depending on Sequence, Mathematics Confidence and Box

Maths Conf.	Box	Sequence 1	Sequence 2	Sequence 3
Lower	Black	N/P	Part. 6 (5, 0.3)	
	Glass	Part. 15 (5, 0.8)[1]		
	Open	Missing audio		Part. 33 (6, 1.0)
Higher	Black	Part. 1 (8, 0.8)		Part. 9 (7, 1.8)
	Glass	Part. 16 (7, 0.6)	Part. 39 (8, 0.4)	N/P
	Open		Part. 30 (8, 1.0)	

[1] Numbers in the parentheses represent the actual mathematics confidence scores and mean performance scores for the participant (out of 2) respectively

All the data (that is, the observational notes, videos, typewritten answers and transcripts) were used together for cross-referencing purposes when trying to explain what the students did. The examples illustrating the steps undertaken by the students were drawn mainly from the 8 students' transcripts. Any noted student behaviour for a group combination (for example, glass-box software and low mathematics confidence) was cross-checked with the observational notes and, if necessary, additional student data were presented for illustration (see for example think-aloud data from Participant 12, p.217).

The think-aloud session was intended to determine students' thought processes or their self-explanations. However, from the transcripts, students' thought processes could not accurately be determined except when they were typing their answers, because when the student read the question in most instances they began typing the answer soon after and creating explanations as they typed. For example:

1:22:56: Clicks Q3C in the answer form and then reads instructional materials. *"Last part. If the number of hours of carpentry per day is increased from 48 to 60 ...",* continues reading out the question.

1:23:20: *"Right ... obviously",* looks at the screen, *"obviously first of all the total production would be increased",* looks down from the screen.

1:23:27: *"So, firstly",* begins to type the answer, *"total production would be increased, overall production ...",* continues typing, *"kinda obvious",* continues typing, *"would be increased and in particular",* stops typing, looks at papers, *" ... they would increase the number of stools produced",* continues typing, *"... would increase the number of stools produced as the solution to the first question, the first part of the question."*

(Participant 16, M, GB, Higher MC = 7).

The codes in Participant 16, M, GB, Higher MC = 7, refer to, firstly, the participants' gender (F = female; M = Male), the second code refers to the software box (BB = black-box, GB = glass-box and OB = open-box) and the last code refers to mathematics confidence (MC), where lower and higher MC are used to refer to mathematics confidence groupings and "= 7" refers to the actual mathematics confidence score. These participant codes are also located in Table 11 (p.91).

Hausmann and Chi (2002) found that students who produced spontaneous self-explanations were inhibited when required to type their self-explanations rather than just thinking-aloud. In their experiment, students studied only the instructional materials rather than solving the tasks when thinking-aloud. In the present study, in contrast, the students did not have to type their self-explanations but were rather encouraged to think-aloud and to type their answers whilst solving the tasks. It is possible that typing may have inhibited the self-explanations that students were able to generate out loud. Further, Gentner (1988) suggested that typing may inhibit cognitive ability and Hausmann and Chi (2002) suggested that the inhibition of their cognitive ability may be a contributory factor in the self-explaining process. However, as students were required to type, they perhaps knew they could change the answer they typed as versus with a pen-and-paper when the answer should be written usually correctly the first time. Students often changed their answers, by either erasing phrases and rewriting new phrases or adding in new words when typing in their answer.

> 46:01: *"This is a shot in the dark … you won't want to have a high value for u".*
>
> 46:09: *"There must be a reason",* Types answer: *"you wouldn't want a high value for u because as u in(creases)",* then erases the last bit to have, *"you wouldn't want a high value for u because as u approaches 90 it would",* then erases the last bit to finally have, *"you wouldn't want a high value for u because".*
>
> (Participant 16, M, GB, Higher MC = 7).

Hausmann and Chi (2002) suggested that having to check grammar or spelling when typing may inhibit students from interacting with the materials in a deep way. In this study, students were told not to worry about spelling and grammar as long as the researcher understood what they were typing. Even so, it was observed that students 'fiddled' with their typewritten answers such as fixing grammar or erasing parts of the sentences, such as Participant 16 (in his answer above). However, although the students typed their answers immediately, it was possible that they were still engaging with their materials. There were two types of students when it came to typing answers: there were those who spoke as they typed their answers and those who typed without speaking. Further, some students were more verbose when it came to thinking-aloud than others and this may be due to personality where people who were more talkative tended to think-aloud better than the quiet personalities. According to Ericsson and Simon (1984) using a practice task should get a student accustomed to thinking-aloud and through prompting, students will be forthcoming in thinking-aloud. However, the practice task used in this study did not seem to overcome the differences in personalities. For example, even when prompting, some students were not forthcoming in their answers (1st example) whilst others were quite verbose (2nd example).

1:00:34: Restarts the tasks, and clicks input problem and inputs the numbers. Clicks iteration and then looks at her instructional materials.

1:02:20: I ask 'What are you looking at?': *"Ah, the solution for number 2 and I'm gonna answer the thing now."*

1:02:43: Enters the answer.

(Participant 1, F, BB, Higher MC = 8)

40:18: After being prompted to talk by saying 'Just keep talking': *"I'm reading equation 2 on page 2. So, that whole paragraph there, I'm just looking for something to wing this problem 2, so I can give you an answer."*

40:59: After being prompted to talk again by saying 'Just keep talking', *"Alright well, then part b, then since they didn't find, what the, what the x, y and t was then I'm guessing if we relate it to the ... well, to page 2, right, of the introduction and it was talking about the farmer, I'm thinking those problems, those… ummm... those variables could actually be like real-life stuff like key ones like how they had it labourers and they could actually for the constraints, they could actually have a value that would equal to something less than or equal to a number because ... because of what? Hold on ... I lost my thought there ... sorry."* Continues looking at instructional materials.

(Participant 15, M, GB, Lower MC = 5)

Since there were different levels of think-aloud, emphasis was placed more heavily on the answers that students typed since, for both the constructive and interpretive tasks, students were asked to give detailed explanations. This was also used to ascertain the explanations they were making. Whilst the typewritten answers did not represent ongoing cognitive processes, they did represent or indicate the sort of self-explanations the students were generating when solving the tasks. The attached Compact Disc (CD) has a compilation of these answers.

The explanations were coded into whether students were using mathematical principles (using any type of mathematics) or relating to real life (using their own practical experience) to solve the tasks (Section 5.4, p.136). The explanation classification helped in determining what direction the students' conceptual thinking was heading. Also, as some of the students began typing their answers immediately, their self-explanations were actually ongoing or formed as they typed and this could be observed by their sometimes fractured typewritten sentences that were corrected or erased, as was seen with Participant 16 (p.296).

6.3 Mechanical Tasks

All 38 students performed the mechanical tasks correctly, as this simply required inputting values and clicking buttons. Further, if there were any inputting errors, the researcher brought this to the attention of the student. Here is an example of the researcher correcting Participant 16 for Problem 3's mechanical task:

> *32:41: "So, I'll input the values into the programme"* inputs the values.
>
> 34:44: He has inputted the wrong thing for the third constraint (inputting numbers from the second constraint) so I ask him which line he is reading and then he reads out the second constraint and indicated to him he entered that before and then he says, *"Oh shucks".* He leaves the variables without any coefficients blank
>
> 35:57: He double checks his values and then tells me*, "I'm just double-checking the values for any errors",* and then continues double checking the values.

In real educational settings, there would not be someone ensuring the correct input of values and errors can arise from misreading, transcription errors or inputting numbers in the wrong order.

Although scores were not collected for the mechanical tasks, it was interesting to investigate how students used the software boxes. This is important as it was determined in Chapter 5 that students' mathematics confidence affected their exploration. This section is divided into sections where the black-box and glass-box software explorations for mechanical tasks are discussed first, followed by the explorations of students using the open-box software.

6.3.1 Exploration Approach: Black-Box and Glass-Box Software

Beyond the 'normal' clicking of buttons to solve the task, only one student (Participant 1) using the black-box software was interested in knowing or understanding the steps, although Participant 9 thought it was 'cool' to just get the answers. Here is Participant 1 transcript on knowing the steps whilst solving the practice task:

> *28:06: "If you don't have to do anything then what's the point ... if you don't have to do anything, if it does it, like what are you supposed to do then?"*
>
> 28:21: I explain it just gives you the answer when you solve it
>
> *28:34: "If it gives you the answer, then you don't really have to do anything. I thought like I would have to do something with these four values and kind of see where it varies and where you're getting the maximum"*
>
> *28:45: "I don't know, if it gives the answer then just ... so that's all you supposed to do with that, click iteration and it gives you the answer"*
>
> (Participant 1, F, BB, Higher MC = 8)

For the glass-box, only 5 students (Participants 13, 15, 18, 21 and 24) tried to conjecture and explain the steps out loud. For example, here is Participant 15 conjecturing what each constraint in an iteration for the practice task represents (the variables *w* and *c* represent the production of wheat and corn respectively):

> 24:09: Clicks iteration. *"Ok, so, I just clicked the iteration button and it came up .. the second set of values came up. I'm seeing Row ... seeing Row 1, well 0, 1, 2",* (the labels for the rows), *"And for Row 0, there is 9000, Row 1 the answer is 40 and for Row 2 the answer is 30. So, I guess [mumbles]",* referring to the objective function and the subsequent constraints.
>
> *24:57: "Well the best, well the answer that the programme said will be the best would be like 2 acres of wheat"* hovers over 2w in the first constraint of the second constraint, *"and no corn"* hovers over 0c in the first constraint, *"and that will give 40 pounds, right? So I believe ... minus s2?"* Hovers over the *s2* column. Scrolls his mouse across the 1st constraint (-½ *s2*) and then the second constraint.
>
> 25:25: Hovers over *s2* column, *"Hold on, I'm checking to see what s2 means"*. Turns to the instructional materials.

23:31: "s2 … slack variable", says whilst looking at instructional materials returns to the screen and then back to the paper.

26:01: "Ok, s2 is the slack variable but it is minus so, I guessing that is for, that is the fertilizer control … and I believe that we would have a minus value for the fertilizer control, can we?" I replied don't know.

26:21: Looks at this instructional materials, *"No, I don't think so you could have a minus value for the fertilizer control … then that would mean, that you already put too much fertilizers and you would cross the amount of 120 that you have and the slack values to actually accommodate for the … to make up the 120. So, having a minus value there would mean that you're going over the limit. So, I think the best one would be number 2."* Hovers over the 3rd constraint (referring to Row 2 I think), *"So, that should be a ½ acre of wheat and 1 acre of corn and that would be the best … and that would be the best one for to make the most amount of money. The best feasible one."*

(Participant 15, M, GB, Lower MC = 5)

There were possibly other students using the glass-box software who were trying to make sense of the steps but it was difficult to know without them uttering their conjectures. The conjectures and explanations of the iterations made by the students did not always correspond to the theory as with Participant 15 above who thought each constraint represented a production of wheat and corn. It was observed that those glass-box students who were trying to understand the steps did this mainly during the practice task and the first mechanical task but the need for understanding the steps for each progressive task decreased. For example, this is seen for both Participants 15 (see Figure 30) and 16 (see Figure 31, p.187).

In fact, most students resorted to using the glass-box like a black-box by clicking the iteration button until they got the answer. Participant 15's (M, GB, Lower MC = 5) transcript whilst he was solving the practice task and subsequently when solving Problem 3's mechanical task illuminates what was occurring (see Figure 30).

Practice Question

24:09: Clicks iteration. *"Ok, so, I just clicked the iteration button and it came up ... the second set of values came up. I'm seeing Row ... seeing Row 1, well 0, 1, 2"*, (the labels for the rows), "*And for Row 0, there is 9000, Row 1 the answer is 40 and for Row 2 the answer is 30. So, I guess [mumbles]*".

24:57: "*Well the best, well the answer that the programme said will be the best would be like 2 acres of wheat*", hovers over 2*w* in the first constraint of the second constraint, "*and no corn*", hovers over 0*c* in the first constraint, "*and that will give 40 pounds, right? So I believe ... minus s2?*" Hovers over the *s2* column. Scrolls his mouse across the 1st constraint (-½ *s2*) and then the second constraint.

25:25: Hovers over *s2* column, "*Hold on, I'm checking to see what s2 means*". Turns to the instructional materials.

23:31: "*s2 ... slack variable*", says whilst looking at instructional materials returns to the screen and then back to the paper.

26:01: "*Ok, s2 is the slack variable but it is minus so, I guessing that is for, that is the fertilizer control ... and I believe that we would have a minus value for the fertilizer control, can we?*" I replied 'don't know'.

26:21: Looks at his instructional materials, "*No, I don't think so you could have a minus value for the fertilizer control ... then that would mean, that you already put too much fertilizers and you would cross the amount of 120 that you have and the slack values to actually accommodate for the ... to make up the 120. So, having a minus value there would mean that you're going over the limit. So, I think the best one would be number 2*". Hovers over the 3rd constraint (referring to Row 2 I think), "*So, that should be a ½ acre of wheat and 1 acre of corn and that would be the best ... and that would be the best one for to make the most amount of money. The best feasible one.*"

27:01: I say to him 'Ok, I'll like your thinking, but it is not exactly where we are getting the answer from. You've got to

Checks what *s2* – suggests not an initial engagement with the instructional materials

Tries to understand the constraints – but incorrectly interprets them – so, to some extent engaging in the problem solving

Researcher indicates to click iteration button to get the solution – this may influence how he works in the future

click iteration until it says the problem has been solved.'
27:14: "*Ok*". He clicks iteration and then I tell him 'Right. Click iteration again' and then the "best solution has been found" pops up. Then I say, 'Right, now say ok'. He clicks "ok" on the pop-up. I say 'The solution is actually … can you see the solution?" He looks at the screen.
27:51: "*Oh well, I'll tell you just now" … "We see that w, the wheat is 20 and corn is 20. But that is planting no wheat or corn. There are no acre …"*. I interrupt him by saying, 'say that back again'.

Problem 3 (Question 1)

36:05: He clicks ok. He hovers over answer form and I tell him 'Oh, could you just go back to input problem, I just realised something, you made a mistake in, constraint B, it is supposed to be 100, less than or equal to 100, you got 1'. "*Thank you, alright gonna put that in*". I tell him, 'Alright that's it'. He now double checks the values.
36:55: Clicks iteration three times until the pop-up comes up, "*And they found an answer.*" He clicks ok. "*There it is. Ok, I see the answer now, z is 105, x is 0, y is 0, t is 15 and u is 90. So, that's the best answer … solution they can get.*" He wants to know if he should go onto part b, and then I tell him he's got to put it into the answer form and wants to know if he can copy/paste it and tell him he's got to type it in which he proceeds to do.

Clicks iteration 3 times without much time spent reflecting during the iterations

Figure 30: Think-aloud transcript of Participant 15 (M, GB, Lower MC = 5) solving a mechanical task for the practice task and Problem 3 using the glass-box

When solving the mechanical tasks, both Participants 15 and 16 initially spent some time looking at the steps in the practice tasks. Participant 15 appeared to engage with the steps more than Participant 16, although Participant 16 was interested in knowing what BV (basic variable) represented [26:54]. However for their subsequent tasks, both Participants 15 and 16 quickly clicked through till they got an answer.

Practice

26:15: "*Right, so, I click iteration to solve the problem*". Clicks iteration. Looks at the new iteration and then looks at instructional materials.

26:37: After being prompted to talk: "*Right, so, that's the iteration and so … ummm…*". Keeps moving his mouse over the iteration.

26:54: "*What's BV again?*" Sees this in the iteration. Looks at instructional materials, "*Basic variable*". Keeps reading his instructional materials.

Wants to understand some aspects of the steps i.e. BV

27:20: "*I guess we click answer form now and then enter the answer*". He clicks answer form. I tell him, 'Ok, not so fast … you got to click iteration' when the answer form pops up, I tell him, 'click cancel for a minute', he clicks cancel on the answer form, and continue telling him, 'you got to click iteration, until you see a pop-up, that tells you the problem has been solved'. "*Ok*". I reply, 'alright'.

27:42: Clicks iteration again, and gets the pop-up that says problem has been solved

Problem 3 (Question 1)

32:41: "*So, I'll input the values into the programme*", inputs the values.

34:44: He has inputted the wrong value for the second constraint (inputting numbers from the second constraint). I ask him, 'Which line are you reading', he replies, "*pardon?*", I say again, 'Sorry, I said which line are you reading?'. He replies, "*Umm…after the second line … 3x + 4y + 60 – u*". I say to him, 'You've just entered that before'. He says "*Oh shucks*". Erases the values and continue entering the values. He leaves the variables without any coefficients blank

35:57: He double checks his values, "*I'm just double-checking the values for any errors*", and then continues double checking the values.

36:11: Clicks ok, "*And now I click iteration*". Clicks iteration thrice, until the solution have been found pop-up comes up.

No longer interested in understanding the steps but just getting the answer

Figure 31: Think-aloud transcript of Participant 16 (M, GB, Higher MC = 7) solving a mechanical task for the practice task and Problem 3 using the glass-box

Both Participants 15 and 16 clicked the iteration button quickly to get an answer (timestamp [36:55] and [36:11] respectively) in their subsequent task that they solved which was from Problem 3. Timestamps for students' transcripts are represented in square brackets in this thesis. The rapidity with which they clicked the iteration button might at first appear to be due to the researcher, who indicated to both of them that they should click the iteration button until the pop-up button appeared ([27:01] and [27:20] respectively for Participants 15 and 16). However, similar instructions had been given in the instructional materials (Appendix 5, p.304).

The students here were treating the glass-box software like black-box software, but not all students did this. Participant 39 was told the same thing during the practice task [21:08] and in the subsequent task that he solved which was Problem 2, what was noted was that after each iteration he looked at the screen and then clicked iteration afterwards [29:53] (Figure 32). There is uncertainty whether Participant 39 was thinking or at least observing what was happening at each iteration step. Similarly, when he did the mechanical task in his second question (Problem 3) (not shown) and his third question (Problem 1) [55:33], he appeared to again be looking at the screen after each iteration, although it was difficult to know for certain since there were internet interruptions. At timestamp [56:25], he made the statement "*I'm just figuring it out",* although the thought was fragmented, it suggested that he was thinking what the answers from the software or linear programming might be representing.

Practice

20:00: "*And I [mumbles] the iteration button*". Clicks iteration button and gets the first iteration, "*Well, I ... uh ... iterate ... ok*", keeps looking at the screen, "*…Is that the answers from before?*" Turns to the instructional materials, "*Yes, it was uh ... z is 10,000*", looks back at the screen, "*Something is amiss here*", keeps looking at the screen, "*I got 9000, 40 and 30*". He turns back to the instructional materials, "*Yeah, that's right, ok*", and then looks back at the screen. I tell him, 'Just a minute, the excel file is saving itself, so it might go blank in a minute'. Excel file saves. I tell him, 'you can go ahead now.'

He checks back the instructional materials for the answer not a definition

21:08: "*What do I do ... reset it now?*". He hovers over the rest button. I tell him, 'No, no. Actually you haven't got the answers as yet. You got to click iteration until it says the best solution has been found'.

Researcher also indicates he must click iteration until he gets the pop-up box

21:18: Hovers over iteration, "*Oh ok ...*", turns to the instructional materials, "*So, the 20 acres, the 20 acres of wheat and the 10000 dollars ... no, ok*", clicks iteration, "*Iteration again*". Looks at the screen, "*Ok that's it ... yeah, that's it*", I tell him, 'just click iteration. Just once more. Just for me to show you this. Just click iteration.' He clicks iteration and the best solution pop-up comes up and I tell him, 'right, that's how you know the best solution has been found', he says, "*The best solution has been found ... Ok ... it come up, the best solution ... that's good ...Ok*", clicks ok.

Looks at the screen after each iteration

Problem 2 (Question 1)

29:53: Clicks iteration. Looks at the screen. Clicks iteration again. Keeps looking at the screen, "*Ok*", clicks iteration, the pop-up comes up, "*The best solution has been found*".

30:23: "*Ok, when I solve it, store it in the answer form?*", I tell him, 'yes'. Clicks answer sheet, "*Ok, z is 140, x is ...*" and then types in the answer

Looks at the screen after each iteration for the problem right after the practice problem

Problem 1 (Question 3)

55:33: Clicks iteration. Looks at the screen. Clicks iteration

Clicks iteration – appears to pause between each iteration

again. Keeps on looking at the screen (not sure if it is because the internet is slow). Clicks iteration and the pop-up comes up, *"Alright"*, clicks ok and gets rid of the pop-up. His mouse hovers over the last column in the last iteration for the answers rather than the separate answer column, then he keeps looking at his instructional materials.
56:25: *"Oh"*, Prompts him to speak as he clicks answer form, *"No, the answer ... [mumbles] ... I just figuring it out ... Mhmmm .... Uh ... pen"*, types in the answer, clicks ok and gets rid of the answer sheet.

Figure 32: Think-aloud transcript of Participant 39 (M, GB, Higher MC = 8) solving a mechanical task for the practice task, Problem 3 and Problem 2 using the glass-box

Further, he hovered over the iteration where it appeared as if he read out the values of the variables from here rather than the separate column where the answers are placed. This suggests he was looking at the iterations and making sense of the new equations being presented to him after each iteration. However, without him explicitly stating that he was looking at these equations meant the conjecture that he was engaging with glass-box software is not conclusive.

The main difference amongst Participants 15, 16 and 39 was that Participants 15 and 16 appeared to lack engagement with the instructional materials as well as with the glass-box software whilst Participant 39 appeared to have made efforts to observe and perhaps understand what was happening whilst using the software. There were thus two ways students were using the glass-box software to solve mechanical tasks. The first way was that some students paid attention to the steps in the practice task but in subsequent tasks dismissed the need to look at the steps and then began treating the glass-box software almost as black-box software. The second way was that the students continued to be interested in observing and looking at the steps even when they had completed four mechanical tasks (that is, the practice and three mechanical tasks from the problem). The former group of students thus had less engagement with the glass-box software and this may reflect a surface processing level when using glass-box software. The second type of student who engaged more or at least paid attention to the screen was probably using a deep level of processing.

6.3.2 Exploration Approach: Open-Box Software

In the open-box software, the solving of mechanical tasks was slightly more complicated than in the black-box and glass-box software since students decided which pivot variable to choose (see Figure 33).

Problem 2 (Question 2)

49:51*: "Ok, let's see what this one is talking about"* read the instructional materials, *"First of all, let's solve it, ok, so we can do that".*

50:18: Looks at the papers and the screen [….]

51:20: *"Let's see if I can pick which one to solve."* [….]

52:06: *"Ok, I'm going with x having the biggest influence, so, I'm going to choose that as my pivot variable",* chooses *x* and gets his first iteration

52:57: Hovers over the column *x* in the new iteration, hover over *y* and then *t*.

53:12: "*The next one I'm going to do is t … which I think is … appears less often, which you want more of*". Hovers over *y*

1st iteration: Decides the PV correctly

2nd iteration: Unable to choose correct PV

Figure 33: Think-aloud transcript of Participant 33 (M, Lower MC = 6) solving a mechanical task for Problem 2 using the open-box software

Although in most cases, students' initial conjectures (that is, choosing the variable that would yield the highest profit) were true, (for example at timestamp [52:06] for Participant 33 in Figure 34) , they often got muddled when presented with the second iteration and were uncertain as to how to proceed (see timestamp [52:57 to 54:49] for an example in Figure 34). Further, when the students using the open-box software had to interact with the software box, they were often concerned about whether they were doing the task correctly and if they were getting the right answer (for example

[55:27 to 56:51] in Figure 34). The students using the black-box and glass-box software did not have these qualms.

Investigating the transcripts, videos and observation notes of students solving the mechanical task with the open-box, it appeared that those with higher mathematics confidence were more likely to spend longer on the tasks and trying to understand the process. This did not mean that students with lower mathematics confidence did not have similar scores, but perhaps tackled the tasks with less engagement.

For example, Participant 33 indicated that he had a lower mathematics confidence but still seemed to understand basic mathematics principles and engaged with the instructional materials to some extent. For example, he understood what the slack variables were when he was presented with the canonical form of the practice problem (see Figure 34, [14:25]). Recall that the canonical form is the basic structure of linear equations, with all the variables on the left-hand side and all numbers on the right-hand side (Section 3.7, p.86). Recall also that slack variables are the variables added to make inequality constraints become equations (that is, have an equal sign).

However, when carrying out the mechanical part of the Problem 1, he wanted to pivot z [26:58]. Recall that the pivot variable is the variable that is increased in order to increase the profit, z (Section 3.7, p.86). The pivot variable is usually the variable with the largest influence on z and the pivot variable is chosen based on its coefficient. As coefficients change from one iteration to the next, this means that the pivot variable will also change. The variable z (the profit) could never be the pivot variable as its value is dependent on the other variables. Thus, Participant 33's logic for choosing z was that he needed a variable that appeared once [26:58] as this was why he had initially chosen x – "*the simplest one*" [26:00]. He further expected that he could not choose y since it appeared in all the equations [26:58]. Here, he was making a conjecture that did not seem to be grounded in the mathematics of the problem.

Practice

14:25: "*Ok, so it makes all the slack values for me … that's handy*".

Problem 1 (Question 1)

26:00: *"So, we work with the simplest one, which is the x one and I'm going to click on iteration with x"*

26:12: Proceeds to click iteration and choose *x*, *"And we'll see what happens"*. The iteration is produced.

26:58: *"I think I want to know if to iterate again and just trying to work out which one basically. Well, y appears in most of the equations. So, don't know if that is a good one to go for. I tried to go for ones that appear once last time. So, I might try and go for the z one this time."*

27:22: Iterates and chooses *z* and realises it says it cannot use that variable choose another. *"Alright, I got to use another one"*

27:59: *"I got to see if there is anything left in there."*

28:25: *"Let's go for the y then"* Chooses *y* and gets an iteration.

Problem 2 (Question 2)

52:06: "O*k, I'm going with x having the biggest influence, so, I'm going to choose that as my pivot variable*", chooses *x* and gets his first iteration

52:57: Hovers over the column *x* in the new iteration, hover over *y* and then *t*.

53:12: "*The next one I'm going to do is t … which I think is … appears less often, which you want more of*". Hovers over *y*

54:00: "*Or I could just try them randomly until I get one*".

54:05: "*Try y*" Chooses *y*. And gets that he cannot uses that variable

54:17: *"Yeah, if I try to do x, y"*

54:23: Chooses *t*, and gets a new iteration.

[…]

55:27: "*I'm just curious to see what would happen if I had*

Chooses x since it is the simplest one

Does not choose *y* because it appears in all – but wants to pivot *z*

Uses *y* when *z* does not work

Chooses *x* since it had the highest influence

Wants to use *t* – as it appears less often

Decides to use *y*

chosen different ways around. So, I'm just going to see if there is a difference, to get rid of these two variables here". Hovers over the *y* column in the canonical form and *t* column in the first iteration […] 56:49: *"I can't really remember what I got the first time around".* 56:51: *"I remember y was zero and therefore the same again … maybe it doesn't matter too much then but I already did it".* **<u>Problem 3 (Question 3)</u>** 1:08:43*: "Uh it doesn't seem to matter which one you pick once it is the right answer … just pick any".* […] 1:10:10: Clicks iteration: *"Cause we have 4 variables this time we need to do one more, so we need to do for u as well",* chooses *u* for the iteration*, "it's a bit bigger",* and gets his iteration.	Tries the pivot variables in a different order Realises the order does not matter The variables and no. of iterations relationship

Figure 34: Think-aloud transcript for Participant 33 (M, Lower MC = 6) whilst solving mechanical tasks using the open-box software

However, when he began to carry out the mechanical task in Problem 2, he made the correct conjecture by using *x* since it had the biggest influence [52:06] but then returned to his previous deduction to use *t* as it appeared less often [53:12]. This did not last long as he changed his mind and decided to use *y*, but, soon realising that this did not work, he reset the mechanical task to its canonical form and then used *t* as his pivot variable [54:05 to 54:23].

The reason why he changed to *y* in this case may be a latent effect from when he did the mechanical task in his first question (Problem 1), where his conjecture of using a variable because it appeared less did not work, but using a variable (*y*) that appeared several times worked [28:25]. Whilst the reason why this was working seemed to confuse him, he was unable to determine a satisfactory reason and persisted in using this conjecture for Problem 2.

One of the things that Participant 33 did wonder about was whether the order of choosing the variables made a difference to the answer he got [55:27 to 56:51]. Hence, when he started Problem 3, he indicated that he no longer had to decide since he could pick any pivot variable [1:08:43]. However, his reasoning for the number of iterations he had to go through was interesting as he indicated that because there were 4 variables he had to do 4 iterations [1:10:10]. Whilst, this was not strictly true, in the case of the linear programming tasks he had to solve, this was a rule-of-thumb that could have worked.

The way in which Participant 33 worked out how to carry out his iterations was quite different from Participant 30 who had a higher mathematics confidence. Participant 30 also understood the canonical form with the practice task, but apparently with a deep level of processing, since from the canonical form of the iterations he understood the basic solution ($z = 0$, $w = 0$, $c = 0$) of the canonical form (see [26:00 to 27:21] in Figure 35).

In the practice task, he proceeded to do the iterations by first choosing *c* and then *w* as his pivot variables. When he started his first question (Problem 2), he chose *x* as his first pivot variable and then proceeded to choose *y*; however, the open-box software indicated through a pop-up box that he could not use *y*. The reason why *y* could not be used was because it had a positive coefficient in the iteration. This prompted him to go back to the practice task to understand what he did previously. He then proceeded to solve the practice task again to ensure he understood what was occurring. Participant 30 finally solved Problem 2's mechanical task by comparing it with the practice task [53:46 to 56:03].

Practice

26:00: […] "*So … ok, this is where z is 0, so we have to find … I think …*", looks back at the instructional materials and reads, "*w and c equal to 0, to find for z … that's*", hovers over *s1*, "*hmmm*", looks back at instructional materials.

26:59: […]"*A step-by-step … well, we have to make w and c 0 first to find*", clicks cancel to close the input variable dialog, "*and after that you have to find a next … well, choose one of the variables … just now … I think I confusing myself, I hope not*".

27:21: Looks at instructional materials again, "*Ok, so … right … everything there. W and c zero … right, so then you can calculate for the rest. Like z, s1 and s2. Like since z is zero now, you have to substitute … ok s1 is 100 and 120 for s2 …. Hmmm … right … ok … so, pivot variable,*"

Problem 2

53:46: Clicks Problem 1 sheet. "*Hmmm … alright so*", hovers over *t* in the canonical form, "*You take that one*", moves his mouse across all of the slack variables, "*this*", hovers over the *s1* column, "*wait just now*", goes back to the practice sheet.

54:07: Moves his mouse over *s1* and *s2* on the practice sheet and then *c* in the canonical form, then *s1*, turns to his instructional materials, "*s1 comes from the equation B … right [mumbles]*", looks back at the screen, "*Ok*", yawns, "*sorry … the equation*", hovers over *c* and *s1* in the canonical form, "*closest to it … [mumbles]*".

54:52: Clicks problem 1 sheet. "*Ok*", hovers over *s1* in the canonical from, "*[mumbles] … ok, so*", clicks iteration, "*this one*", […] Clicks ok and gets the second iteration.

55:29: "*Ok, umm … we have to do this here*", keeps looking at the screen, "S*o, it's c for t … right*", hovers over the *s1* and *s2* columns in the canonical forms, then hovers over the *t* column.

Understood the canonical form and basic solution

Corresponds between the practice sheet and his Problem 1 sheet to understand solving the problem

Makes a relationship between the *c* in the practice sheet to the *t* in his Problem 2 sheet

56:03: Clicks iteration and gets the pop-up that the problem has been solved, "*That's it?*", I confirm, "Y*ou for real? I, I … nah, I do something wrong*".

Problem 3

1:29:04: After being prompted: "*I was trying to get …umm … the different variables but I only get the same answers all the time*", laughs, "*Right, just now … I could regroup*", laughs, *"right … so*", moves his mouse along the column *t* for all iterations, clicks practice sheet.

[…]

1:35:47: After being prompted: "*Well, first I was looking for the umm … uh … the highest yield, I was looking, well, the first variable, the first equation, negative 6, negative 8 and negative 13. And I was just thinking of the number line and choosing negative 6 as the highest … or is that a wrong line of thought?*", I tell him I cannot comment, "*Ok, right, yeah so I was thinking about that being the highest, you know, to start it off with that variable and each one, every … well, after I iterate a value, the negative 6, I got other equations and substituted a next variable from the umm the resulted equations, I got another set of equations that giving me the same values, you know*".

Getting the same answer for the iterations

His conjecture on using the iteration except uses the lowest negative number rather than the highest

Figure 35: Think-aloud transcript for Participant 30 (M, Higher MC = 8) as he solves mechanical tasks using the open-box software

When he started doing his second question (Problem 3), he felt he was on the correct track but realised that, when he did the iterations, the values on the right-hand side were not changing. He hence doubted whether he was performing it correctly and decided to check the practice task again [1:29:04]. He was actually using an almost correct conjecture concerning how to solve the task [1:35:47]. However, he was using the highest coefficient in the objective function in the number line, rather than the highest negative number. The highest negative coefficient would have represented the variable with the highest profit. Hence his conjecture was not correct, and he started doubting himself when it came to using the software box. Thus he was choosing the value -6 rather than -13, because -6 was the highest coefficient on the number line. When he was carrying out Problem 1, he did not have this difficulty, since in that task there were only two negative variables and choosing either one would have produced an answer quite readily using the open-box software.

Participant 30 was interesting, because he seemed to be using a worked-out example. Chi *et al.* (1989) indicated that poor students may look back at instructional materials. They noted that using worked-out examples was a phenomenon that occurred for both Good and Poor students. However, Good students looked at worked-out examples differently, in that they used them for a specific reference. Thus Participant 30 probably was using the practice task as a reference to understand what was occurring in the other mechanical tasks he was solving. However, Participant 33 did not look at his practice task, because once he realised that the iterations would always give him the correct answer he had no need to understand what or why something was occurring. Therefore, it appeared that Participant 30 probably had a deeper processing level than Participant 33 when it came to using the open-box software.

Therefore, looking at how students were approaching the mechanical tasks, it seems that students using the black-box software were not concerned with understanding the calculations or steps as they were never presented to them. However, the students using the glass-box software might be interested at first in the steps but not after carrying out subsequent tasks. The students who were not interested in the steps tended to be in the lower mathematics confidence group (Participant 15 was a notable exception) and were probably less likely to use a deep processing level for observing the steps. Although only 2 participants were presented here for the open-box software, the illustrative examples show how a higher mathematics confidence student was able to have cohesive thoughts on when using the software box unlike the student with lower mathematics confidence who demonstrated fragmented mathematical thoughts, even though both scored similarly on the tasks.

6.4 Interpretive Tasks, Performance and Approaches

In Chapter 5, there were several important findings with respect to the interpretive tasks. Firstly, students who had higher mathematics confidence did better in the interpretive tasks than those with lower mathematics confidence (Section 5.5.1, p.139). Secondly, for the interpretive tasks, all students performed better in Problems 2 and 3 than in Problem 1. This section provides insights into these findings by discussing performance and mathematics confidence. The impact of software boxes on the interpretive tasks is discussed in Section 6.5 (p.207)

6.4.1 Performance and Mathematics Confidence

Some typical answers by students for the interpretive task in all three problems depending on their mathematical confidence are presented in Table 35.

In the quantitative data it was found that students with higher mathematics confidence tended to obtain higher scores than those with lower mathematics confidence across all tasks (Finding 3 in Annex 5, p.326) but particularly for interpretive tasks (Finding 7 in Annex 5, p.326). By examining the answers between the higher and lower mathematics confidence students, there appeared to be longer typewritten explanations for the answers. This is not true for all students, for example, Participant 12 (F, BB, Lower MC = 2) gave quite verbose answers for all her problems, but these were not always correct.

Table 35: Answers by high and low scorers for the interpretive Task depending on Mathematics Confidence

Interpretive Task	Lower Mathematics Confidence	Higher Mathematics Confidence

Interpretive Task	Lower Mathematics Confidence	Higher Mathematics Confidence
Problem 1		
Low Score	*The solution states that more time is spent on painting rather than carpentry. In order to maximize profits more time should be spent on carpentry hours.* (Participant 20, F, GB, Lower MC = 5)	*From the values of the calculations it is clear that it is much easier to make toy trains than carve toy soldiers. So to maximise profits would mean to divert the some of the painting hours to that of carpentry hours to create a balance in the time spent on each item. As well as they can then increase the overall hours of work.* (Participant 39, M, GB, Higher MC = 8).
High Score	*To make the most profit twice as many trains should be made as soldiers. Carpentry is not being maximised. Painting being run at full.* (Participant 33, M, OB, Lower MC = 6)	*Make 40 toy trains and 20 toy soldiers* *80 hours painting toy trains and 20 hours painting toy soldiers* *40 hours making trains and 20 hours making soldiers* (Participant 9, F, BB, Higher MC = 7)
Problem 2		
Low Score	*Chairs were not produced. demand for chairs was low* (Participant 6, M, BB, Lower MC = 5)	*The chairs. The demand for the chairs was very low so it was not profitable to manufacture.* (Participant 27, F, OB, Higher MC = 8).
High Score	*No chairs were produced* *Not enough time, there wasn't a high demand for chairs.* (Participant 32, F, OB, Lower MC = 5)	*Chairs were not manufactured. This may be because they require more of the limited resources for their products. Time and material available can more easily*

Interpretive Task	Lower Mathematics Confidence	Higher Mathematics Confidence
		produce desks and stools. (Participant 18, F, GB, Higher MC = 8).
Problem 3		
Low Score	*The constraints are always set at minus or equal to a number because the variables usually represent products and it is impossible to produce a negative product. Cannot tell.* (Participant 35, F, OB, Lower MC = 3)	*Maybe the values are of a continuous variable. Hence all the less than or equal to constraints that a problem might naturally have may not accurately be converted to strictly less than constraints.* *x and y should be small because of the constraints.* (Participant 1, F, BB, Higher MC = 8)
High Score	*Constraints are usually the limiting factor in the maximization of the profit. Therefore, the max value of the constraint can be utilized. Rather than a value less than it. This would help to increase profit. Will not want to have a high value for u since this variable would decrease the maximum value of z...from the equation (z = 6x + 8y + 13t - u)* (Participant 19, F, GB, Lower MC = 6)	*Solution:* *We allow <= constraints in linear programming rather than just < because sometimes we may be interested in solutions on the boundaries of the given problem.* *We will not want to have a high value for u since all the variables are non-negative and in determining z, each of the terms besides the term in u take positive values whereas the term in u takes negative values, thereby the contribution from this term reduces the value of z. hence we would want to keep this*

Interpretive Task	Lower Mathematics Confidence	Higher Mathematics Confidence
		term small in adding the terms to determine z. (Participant 25, M, OB, Higher MC = 8)

The students with higher mathematics confidence appeared to engage more with the material as they were explaining why things were occurring. For example, in Problem 1's interpretive task, students were required to explain what the solution from the mechanical task meant for the toy company. The solution to the interpretive task was that 40 toy trains and 20 toy soldiers should be manufactured to make a profit of £140 and that this manufacturing combination would ensure that demand for the market was met and that all painting hours were used with 20 hours of carpentry remaining.

From Table 35, in Problem 1 of the low scorers, whilst both the lower and higher mathematics confidence students' answers were inaccurate, the higher confidence student tried to explain why things were occurring or at least provide a more elaborate explanation: "*So to maximise profits would mean to divert the some of the painting hours to that of carpentry hours to create a balance in the time spent on each item*". Here the student is suggesting that the painting and carpentry hours should be balanced. The lower mathematics confidence student made statements and gave no explanations even though detailed explanations were asked for: "*to maximize profits more time should be spent on carpentry hours*".

Chi *et al.* (1989) found that high scorers were students who provided more spontaneous self-explanations from which they gained additional understanding of the underlying principles. They indicated further that poor scorers had limited self-explanations; when they did self-explain, their explanations were disconnected from the underlying principles and concepts. For example, Participant 12 who had lower mathematics confidence (Lower MC = 2) was quite verbose but her answers were not connecting with underlying mathematics principles:

> *The equation gives an insight into how the manufactures should deal with their capital in order to maximise their profits at a minimum cost. Here we see that the maximum number of hours for both the painting and carpentry processes are limiting factors. Since the job requires a maximum of 100 for painting and 80 for carpentry and this is as essential as the number of toys produced at that time, then it would be essential to understand the where the trade off should lie between the cost of production and the cost for production. Hence using the equation the manufactures are able to pre-determine which one of these factors should be sacrificed and by how much to develop the required trade off for maximum profits. From the answer acquired the farmer can be more than able to meet his overall objective by producing either of the toys because it would simply take 80 toy trains and 20 toy soldiers to get him to a profit*
>
> (Problem 1's interpretive task, Participant 12, F, BB, Lower MC = 2)

In fact, she was paraphrasing (cf. Hausmann and Chi, 2002) her typewritten answer by describing what the equations meant, rather than making any connections to the underlying mathematics principles such as finding implications of what this meant for the manufacturer.

Perhaps the reason higher mathematics confidence students were scoring higher, was that they were better at using their relational understanding in these tasks. Interpretive tasks require the use of conceptual knowledge. Therefore, when students engaged with the materials they provided self-explanations as to why the solutions were occurring by using a deep level of processing. The students in the lower mathematics confidence bracket were providing typewritten answers that lacked explanations, and it may be that they were using surface processing. Note that this conjecture may appear not to be supported by the findings in Chapter 5, where the frequency of explanations was found not to be influenced by mathematics confidence. However, in Chapter 5 the results applied to the number of explanations and type of explanations (real-life or mathematical). Here what is being suggested is that the quality of their explanations (that is, the detailed content of the explanations) was related to their level of mathematics confidence.

6.4.2 Interpretive Tasks: Performance and Approaches

Whilst the previous section showed why students with higher mathematics confidence performed better in the interpretive tasks, the question still remained why students performed better in Problem 2 and 3's interpretive tasks than Problem 1 (Finding 4 in Annex 5, p.326), and also why students performed better overall in Problem 2's interpretive task (Finding 4 in Annex 5, p.326). The reason seemed to be dependent on the nature of the interpretive task.

For Problems 1 and 2, the first part of their interpretive task required the students to match the variable with the number found in the mechanical task. For example, in the first part of Problem 2's interpretive tasks, students had to note that the variable y referred to chairs, and hence the solution of $y = 0$ in the mechanical task meant that no chairs were produced.

Problem 3's interpretive task was different from the two previous interpretive tasks, as the students had to infer a reason from the instructional materials as to why less-than and equal-to constraints were used rather than only equal-to constraints. Also, students were asked to identify which variable should be minimised. The second part of both interpretive tasks requiring the students to explain their answers. This was conveyed by asking students 'why?' in Problem 2 and 3's interpretive tasks, whereas in Problem 1's interpretive task students were only asked to interpret and give detailed explanations. Problem 2 and 3's interpretive tasks are discussed first, and then comparisons are drawn with Problem 1's interpretive task.

For Problem 2's interpretive task, all but one student was able to perform the first part correctly, which is one reason why Problem 2's interpretive task had a high score. In Problems 2 and 3, students had low scores in the second part (that is, answering the why part of the interpretive task). In Problem 2's interpretive task students were often able to give an explanation, but it was usually an incorrect one. Students would note the constraint $y \leq 5$ (the demand for chairs) when examining the set of linear inequalities and then deduced that this was the reason chairs (y) were not being produced (see Table 35). However, if students examined the solution, they could have noted that the value of 5 was well within range for desks ($x = 2$) and stools ($t = 8$).

It seemed as if this was an answer to which they quickly connected without a thorough engagement with the material. Students with lower mathematics confidence were mostly satisfied with saying that the demand was low, whereas higher mathematics confidence students were more likely to realise that it was because chairs required more resources to make (7 higher MC vs 3 lower MC students). In the case of Participant 13 (M, GB, Higher MC = 8), after deciding that the demand should not have any effect, he confirmed this on his glass-box software. He removed the demand for chairs (y) constraint in the software box by changing all the coefficients on the left hand side and the value of the constraint on the right hand side to zero for this constraint.

In Problem 3's interpretive task, students were often stumped and at least 7 students wrote "I don't know" for the question, albeit mostly for the second part of the question rather than the first part. Whilst in Problem 2, a guess could be hazarded, students were unable to make an educated guess, perhaps because of the abstract nature of the interpretive task. Hence any explanation had to be related to mathematics principles. Whilst both lower and higher mathematics confidence students chose to use mathematics principles to explain these tasks, it was the higher mathematics confidence students who used the relevant mathematical principles to ensure that they were typing the correct answer. The lower mathematics confidence students tended to have problems with algebra (see below). Since the higher mathematics confidence students were more likely to explain according to relevant mathematics principles (as seen in Table 35), this was perhaps why the higher mathematics confidence students performed better in this task.

However, in Problem 1, whilst there was a similar read-out to Problem 2, (that is, corresponding variables to calculated values), students were not able to match the values as quickly, possibly because students seemed uncertain about how to answer "interpret what this solution means". Often instead of interpreting the solution with respect to the variables and associated numbers, they tried to explain what the constraints were saying (see Table 35), rather than pointing to the fact that the solution meant that the toy company had to produce 20 soldiers and 40 trains. Perhaps, to the students, this was implicitly stated and no longer needed to be reiterated.

Students of both mathematics confidence levels tended to paraphrase or explain the set of equations rather than trying to understand the implications of the solution to the manufacturer and this was reflected in their answers. Thus, both set of students performed poorly in this task, although the higher mathematics confidence students performed slightly better. The underlying question here is what caused the students to paraphrase rather than self-explain? One of the interesting things about this task was that whilst it asked the student to explain what this solution meant to the manufacturer, it did not include the explicit key word 'why?'. Hence students may have decided that this meant presenting things in a paraphrased manner, and so they simply described what was happening. However, in tasks that explicitly used the word 'why?', students were prompted spontaneously to self-explain.

Further, some students appeared to have poor algebra skills in interpreting Problem 1's solution correctly as they seemed to have a misunderstanding about variables and coefficients. The students found and wrote the solution to the mechanical task as $x = 40$ and $y = 20$. However, when these students explained what the number of toy trains (x) and toy soldiers (y) were, they seemed to mix up the profit (from the objective function: $z = 2x + y$) obtained by the toy trains and toy soldiers with the amount produced. These students became confused and indicated that the number of toy trains (x) being produced was 80 (that is, 2×40). For example:

> *The maximized profit z, was calculated to 100, where the no of toy soldiers manufactured is 20 and the no. of toy trains manufactured is 80.*
>
> (Participant 10, F, BB, Lower MC = 5)

At first, this confusion appeared to be due to the lower level of mathematics confidence. However, students in both mathematics confidence groups made similar errors, although the frequency of errors was higher for the low mathematics group. Further, the students' algebra pre-test showed that there was no significant difference in scores between the students of higher and lower mathematics confidence. It was possible that students' procedural skills in doing algebra were not affected by their mathematics confidence but that their conceptions of the algebra (in particular, their understanding of coefficients and variables) were.

In short, students found it easy to match variables with answers; however they were apparently able to spontaneously self-explain more if the task asked 'why?'. This could possibly be a good prompt in helping students to self-explain. The higher mathematics confidence students used this prompt more effectively as demonstrated by the higher performance scores from their explanations. Further the students answering the interpretive tasks were affected by their wrong conception of variables and coefficients in algebra.

6.5 Interpretive Tasks and Software Boxes

As noted in Section 5.5.2 (p.144), students using the glass-box software performed marginally significantly ($p < 0.08$) better in the interpretive tasks than those using the black-box software (Finding 8 in Annex 5, p.326). Looking at Figure 23 (p.144), one would note that the scores obtained for the interpretive tasks using the glass-box and open-box software were similar and larger than those using the black-box. As noted in Section 5.5.1 (p.139), the interpretive task scores were dependent on the mathematics confidence of the students and it was conjectured that the higher mathematics confidence students were making better self-explanations and hence using a deep processing level.

There was no quantitative evidence to suggest there was an interaction between the effects of mathematics confidence, problem and software box for the interpretive task. The qualitative data are used to shed some light on why the students with glass-box and open-box software were performing better. Before triangulating with the qualitative data, the quantitative data are examined to determine where the differences may be occurring, even though they are not significant. In Table 36, the scores of the students with lower and higher mathematics confidence and their software box are given for the three software boxes. The maximum score was two. Unusual trends (although not significant) in the data are highlighted in bold and each problem’s interpretive task is now looked at separately.

Table 36: Mean interpretive task scores for the software Boxes depending on the Problem and Mathematics Confidence

Problem	Black	Glass	Open
Low Mathematics Confidence			
Problem 1	**0.25**	0.50	0.83
Problem 2	1.13	0.89	1.25
Problem 3	0.38	**1.17**	0.58
High Mathematics Confidence			
Problem 1	0.72	0.75	1.00
Problem 2	1.39	1.75	**1.00**
Problem 3	1.11	1.50	1.42
Total			
Problem 1	0.49	0.63	0.92
Problem 2	1.26	1.32	1.13
Problem 3	0.74	1.33	1.00

6.5.1 Problem 1's Interpretive Task

In Problem 1's interpretive task, it was noted in Table 36 that students using the black-box software with lower mathematics confidence had a lower score than those students using the glass-box and open-box software.

For Problem 1's interpretive task, the students were required to explain what the solution of z = 100, x = 40 and y =20 meant to the toy manufacturer. The students were supposed to match these values to what the variables represented: that is, they were expected to write an answer such as "The toy manufacturer will make a profit of £100 if they manufacture 40 toy trains and 20 toy soldiers". There were additional statements that they could have made with respect to the constraints, such as whether the toy manufacturer had met the market demand for toy trains and also whether the toy manufacturer had used all the carpentry and painting hours available.

Examining the answers, which were provided by the students using the glass-box and open-box software versus those students using the black-box software, yielded an interesting observation. All but one of the lower mathematics confidence students using the black-box software explained the solution in terms of the constraints rather than by matching the values to the variables. For example, Participant 37 in Table 37 restated the painting constraint (*2 trains and one soldier in 100 painting hours*) and then suggested from this constraint what the manufacturer could do to increase his profit (*decrease the time it takes to produce a train*). On p.202, there is also an example of Participant 12 (F, BB, Lower MC = 2) describing constraints for this interpretive task.

This was also seen for students using the glass-box software (for example, Participant 19) but to a lesser extent as most students appeared to match the values to the variables even though sometimes it was not explicitly stated. However, most students assigned to the open-box software were likely to interpret the solution as being 20 toy trains and 40 soldiers manufactured or, if not explicitly stating the solution, they provided insight into how this production would affect the constraints (see also Participant 33 in Table 35, p.198).

These results suggest that the glass-box and open-box software may encourage students who have lower mathematics confidence to associate the values with the variables compared to students using the black-box software. Given more time and resources, it might be possible to substantiate this using a larger number of participants.

Table 37: Answers to Problem 1's interpretive task for low mathematics confidence students depending on their assigned software box

Black-Box Software

the max z stand s at 2 trains and one soldier in 100 painting hours a reasonable suggestion would be to decrease the time it takes to produce a train in such a manner that another soldier or train can be made within the carpentry hours, increase the carpentry hours or increase the painting hours possible by 20 hours so as to increase the no. of soldiers painted per day.

(Participant 37, M, BB, Lower MC = 6)

more trains can be made in the allotted hours for both carpentry and painting than soldiers

(Participant 6, M, BB, Lower MC = 5)

Glass-Box Software

Producing 40 trains and 20 toy soldiers gives the max profit

(Participant 17, M, GB, Lower MC = 6)

the amount of hours available for painting is 100 hrs
the amount of hours for carpentry is 80 hrs
the toy train takes twice as long to paint when compared to the toy soldier...
they both take the same time to make
therefore, in terms of labour, the toy soldier would be more profitable
2x + y
means that the train earns twice as much profit as the soldiers...
no of trains has some constraint (c)

(Participant 19, F, GB, Lower MC = 6)

Open-Box Software

Maximum profit is attained by producing 40 toy trains and 20 toy soldiers. The number of hours spent painting and on carpentry work was equal to 20 hours each.

(Participant 35, F, OB, Lower MC = 3

it takes twice the amount of hours to paint toy trains than toy soldiers
while it takes the same amount of carpentry hours

(Participant 31, F, OB, Lower MC = 5)

6.5.2 Problem 2's Interpretive Task

Table 36 (p.208) showed that the higher mathematics confidence students using the open-box software were performing more poorly than their counterparts using the black-box and glass-box software for Problem 2's interpretive task. To further examine this trend, the scores obtained from the interpretive task were decomposed into their two parts (see Table 38). It was noted that whilst the higher mathematics confidence students with the open-box software were able to obtain the correct answer that chairs were not produced, all were unable to produce a reason from which they would gain a score. They instead stated that the demand for chairs was low as discussed in Section 6.4.2 (p.203).

It, therefore, seemed that the higher mathematics confidence students using the open-box software were presenting answers that would have been typically that of lower mathematics confidence students. Examining the answers provided by the open-box higher mathematics confidence students (Table 39), what was noted, as with most higher mathematics confidence students, is that they put forward answers that had lengthier explanations, a phenomenon which was discussed in Section 6.4.1 (p.198).

Table 38: Frequency of scores depending on Mathematics Confidence and software Boxes for the second part of Problem 2's interpretive task

Scores	Black	Glass	Open
Lower mathematics confidence			
0	3	8	4
0.5	1	0	1
1	0	0	1
Total	**4**	**8**	**6**
Higher mathematics confidence			
0	5	1	6
0.5	1	0	0
1	3	3	0
Total	**9**	**4**	**6**

Table 39: Responses by higher mathematics confidence students using the open-box software for Problem 2's interpretive task

Chairs were not manufactured because of its low demand
(Participant 28, OB, M, Higher MC = 7)
product not produced = chairs *reasons being: maybe,* *1) the no. of hours available for carpentry were not sufficient* *2) the feet of lumber available were not enough* *3) the demand for chairs was not great therefore they were not a priority to make*
(Participant 30, OB, M, Higher MC = 8)
According to the solution given by the program the # of chairs (y) = 0 i.e. no chairs have been produced, due to the constraints and in order to gain a profit of 140 the number produced had to be 0 for the desk and stools brought the profit desired.
(Participant 34, OB, M, Higher MC = 7)
No chairs were produced. Maximum was 140 so according to the equation *when x=2, and t=8, y=0 gives the maximum profit.* *Z= 140 also y is less than or equal to 5*
(Participant 29, OB, F, Higher MC = 7)

Some of the higher mathematics confidence students using the open-box software (but not all: for example, Participant 28) appeared to be answering this task with a mathematical bias, in that they were using mathematical symbols. This was in contrast to students using the black-box and the glass-box software who tended to use more prose (Table 40). This did not mean that black-box and glass-box software students did not use mathematical symbols, just that they tended to use them less.

Table 40: Responses by higher mathematics confidence students using the black-box and glass-box software for Problem 2's interpretive task

Program advises me not to produce any chairs since the demand for chairs is not very high. I will not get a high profit from manufacturing chairs and would be wiser to produce more stools and desks since there is a demand.
(Participant 7, M, BB, Higher MC = 7)
Chairs were not produced. Maximum profit that can be obtained from the maximum number of chairs that can be produced is less than the profit that can be gained from the iterated number of stools produced.
(Participant 5, BB, M, Higher MC = 8)
From the answer to the previous question where y=0 no chairs were produced which may be as a result of poor demand which is less than 5
(Participant 16, M, GB, Higher MC = 7)
Chairs were not produced due to the low demand of chairs as well as the amount of lumber available was preferred to make desks and stools.
(Participant 39, M, GB, Higher MC = 8)

It seemed that for students using the open-box software that doing the procedural steps may have made the students more inclined to present their answers in a mathematical pattern as can be see from Participant 30 (M, OB, Higher MC = 8):

> 1:04:11: After being prompted, *"This umm … this answer for this part B, when they say as detailed as possible, you need a detailed mathematical explanation or just a normal detailed explanation?",* I tell him 'as detailed as you interpret it to be'. *"Ok, because I could interpret this to be mean a lot of things".*

He had divided possible explanations into two types, mathematical and other, which is interesting as this is the same distinction proposed in the present research (Section 5.3, p.131). Note however that the mathematical use of symbols would not necessarily be coded as a mathematical explanation. Also, from the quantitative results, on the interpretive task, students who used mathematical explanations were also likely to use real-life explanations (Section 5.7.3, p.159 and Finding 29 in Annex 9, p.332), if they scored a mark. Perhaps, the students with the open-box software tended to latch onto using mathematics to answering the interpretive task; whilst this task could be answered mathematically, perhaps a practical and mathematical approach to determining why stools and desks were chosen as opposed to chairs might have served them better. Instead, the students tried to justify the reason for achieving the maximum £140 profit and indicated that this could only happen if $y = 0$ (Participants 29 and 34), which mathematically works out correct for the equations. Hence, it seems from their explanations, they were not taking a surface processing level but rather were asking themselves the wrong question when they were self-explaining, and perhaps this is where prompted self-explanations could be quite useful (see Chi *et al.*, 1994).

In this case the open-box software seemed to influence the manner of answering the task for the higher mathematics confidence students. In particular, the open-box software seemed to encourage the students to type their answers from a more mathematical view. These students were providing their answers as they understood and saw the inequalities within the iterations presented in the software.

6.5.3 Problem 3's Interpretive Task

To understand why students using the glass-box software obtained higher scores in Problem 3's interpretive task, the scores obtained from this task (see Appendix 6, p.313) were once again decomposed into their two parts (see Table 41). A higher percentage of students using the glass-box software answered correctly the first part of this interpretive task (that is, "*Why do we allow linear programming to have ≤ constraints rather than just < constraints?*") regardless of mathematics confidence levels.

Table 41: Frequency of scores by software Boxes and Mathematics Confidence for the first part of Problem 3's interpretive task

Score	Box			Total
	Black	Glass	Open	
Lower mathematics confidence				
0	2	1	4	**7**
0.5	1	0	0	**1**
1	1	8	2	**11**
Total	**4**	**9**	**6**	**19**
Higher mathematics confidence				
0	3	1	2	**6**
0.5	3	0	0	**3**
1	3	3	4	**10**
Total	**9**	**4**	**6**	**19**

The reason for this was not apparent instantly. However, upon examining the 8 transcripts, it was noted that 2 of the 3 glass-box software students (Participants 15 and 16) referred to the section related to linear programming in their instructional materials before answering that part of Problem 3's interpretive task. Upon cross-examining the observation notes with the videos, it was noted that 3 additional students using the glass-box software also cross-checked with the instructional materials before typing in their answers (Participants 19, 22, 38). One student using the black-box software (Participant 12) and two using the open-box software (Participants 26, 36) also did the same. Further, except for Participant 16, all of the students were grouped into the lower mathematics confidence bracket. Participant 12 who used black-box software and Participants 26 and 36 who used open-box software were the only students of lower mathematics confidence who got it fully correct. Students in the higher mathematics confidence group generally did well in this interpretive task regardless of the software box that they were using.

At first this high interpretive score for the glass-box software students suggested that these students were being more conceptual and drawing deeper self-explanations and hence connecting their knowledge to what was read. However, Chi *et al.* (1994) found that high self-explainers tended to refer to the text a lot less than the low-explainers when they were solving problems. They suggested that the high self-explainers may know the "knowledge inference". Chi *et al.* (1989) also indicated that successful problem solvers were the ones who referred to examples more quickly and usually were goal-oriented in finding their answers. Although most of the students using glass-box software checked the instructional materials, it cannot be suggested that the glass-box software influenced the way the students were learning during the reading of the instructional materials as they were only presented with the software box after reading the instructional materials. All students read the instructional materials first and were then presented with the type of software. So were the students using the glass-box software more conceptually minded because they got this interpretive task part correct, or were they using a surface processing level and had to return to the instructional materials to find the answer?

As mentioned in Section 6.4.2 (p.203), there were 7 students who answered "I don't know" to Problem 3's interpretive task. Examining this further, only three students provided this answer for the first part of Problem 3's interpretive task and all were black-box software students. As this first part of Problem 3's interpretive task yielded low scores for the black-box software students, it was perhaps something in the glass-box software that cued the students to look back to the materials. Although Participant 12 who used the black-box software also looked back at the instructional materials, it was unlikely that this was due to a cue in the software, because of what she said just after reading out Problem 3's interpretive task:

2:55:32: *"Just now I saw it …",* begins turning through the instructional materials, *"hmm … something couldn't be less than something and … yes … it's at the beginning."*

What she meant by "*just now I saw it*" was that, while responding to Problem 1's interpretive task, she looked back at the instructional materials to try and understand how to explain the Problem 1's interpretive task, and hence perhaps this was the reason that she remembered it.

Participants 15 and 16, who used the glass-box software, both returned to the same example or place in the instructional materials (that is, the constraint on labourers):

40:18: After being prompted to talk: *"I'm reading equation 2 on page 2. So, that whole paragraph there, I'm just looking for something to wing this problem 2, so I can give you an answer."*

40:59: After being prompted to talk, *"Alright well, then part b, then since they didn't find, what the, what the x, y and t was then I'm guessing if we relate it to the … well, to page 2, right, of the introduction and it was talking about the farmer, I'm thinking those problems, those ummm, those variables could actually be like real-life stuff like key ones like how they had it labourers and they could actually for the constraints, they could actually have a value that would equal to something less than or equal to a number because … because of what? Hold on … I lost my thought there … sorry."* Continues looking at instructional materials.

41:51: *"Ok, for the twelve, ok for the linear programming to actually have the less than or equal to constraint rather than just the less than constraint because the value you're leading out, that would equal to the … you could have a value that would be equal to an answer, to the best possible answer. You could have that possibility. So, you need to include, that equal value to not just the less than. So, that's why I'm thinking, that value could be important in some cases, so that's why they need to include it in the programming also."*

(Participant 15, M, GB, Lower MC = 5)

39:21: *"Well, I guess they would the constraints with less than or equal to rather than just less than because the boundary constraints … "* looking at papers, *"like in the sample question where you know, they couldn't, well, they only had 100 labourers so* *you couldn't go pass 100 labourers, so, the boundary would be less than or equal to. Cause you have available 100 workers so you can utilize the whole 100 you know, equal to 100 instead of just less than a hundred … cool, so uh … now to put it into words".*

(Participant 16, M, GB, Higher MC = 7)

A possible reason is that this was where the term constraint was first introduced. Thus, it was probable the idea of what a constraint was prompted the glass-box software students to return here. After seeing all the iterations in the glass-box software with the various constraints changing and perhaps not paying attention to what the constraints meant, it prompted the students to realise they did not quite understand what a constraint was and hence returned to the definition of a constraint. Once they returned to this section, they began to self-explain for themselves why this was occurring (see [40:59] to [41:51] for Participant 15 and [39:21] for Participant 16). Both Participants 15 and 16 perhaps used a surface processing level when reading the instructional materials as the former did not know what slack variables were after it was mentioned several times in the instructional materials (see [25:25] to [25:31] in Figure 30) whilst the latter did not know what 'BV' stood for (see [26:54] in Figure 31). Both basic variables and slack variables were defined in Section 3.7 (p.86). Therefore, it seemed that the glass-box software was not promoting a conceptual way of thinking; rather, if students started with a surface processing level it persisted in their use of the software, particularly if they had lower mathematics confidence.

Possibly the same thing happened for the two students using the open-box software (Participants 26 and 36). From their videos and observation notes, it was noted that these two students were choosing pivot variables in solving the mechanical tasks to only get a new iteration. From their think-aloud session this implied they were doing this without trying to understand what was happening and hence were probably using mostly surface level processing. Again, as previously noted the reason that the students seemed to spontaneously self-explain was because of the cue ‘why?’. Thus, whilst the students appeared to be using a surface processing level with the software boxes, by asking them ‘why?’ in Problem 3’s interpretive task, the cue prompted them to self-explain why this phenomenon was occurring. Therefore, the software boxes were not themselves promoting a deep or surface processing level. Rather, the processing level that the students adopted to study the instructional materials was manifested in their use of the software boxes.

Therefore, these results imply that those students who wrote the correct answer without referring back to the instructional materials had already made these knowledge inferences. Also, they were aware of what was in the instructional materials (Chi *et al.*, 1994) and hence relied on their own wits in answering this task.

For the second part of the interpretive task, where students were asked “*which variable will we not want to have a high value for?*”, 15 students with higher mathematics confidence were able to obtain a mark. Only 5 students with lower mathematics confidence were able to obtain a mark. From Table 35 (p.198), the answers by students with higher and lower mathematics confidence illustrate where the differences lie. Students with the higher mathematics confidence were more often able to note by examining the objective function that the variable ‘u’ was negative and hence by having a large value for u will reduce the value of z (the profit). Most students with lower mathematics confidence were unable to note this. Students who were unable to state u as their answer, provided any of the other variables (such as x, y or t) or simply noted ‘I don’t know’. Three students suggested that a constraint (particularly Constraint A) should not be increased rather than indicating a variable. This suggests a poor conception of the difference between a constraint and a variable.

Initially it was thought that perhaps students using the open-box software and, to a lesser extent, students using the glass-box software should be more likely to notice that *u* was the answer since they were going through the iterations. However, students had similar answers for this part of the task regardless of software box used. Again these results confirmed that students with higher mathematics confidence are better able to make sense of the interpretive tasks than the students who have lower mathematics confidence.

6.6 Constructive Tasks

This section looks at the constructive tasks. As the correct solution of the constructive tasks was dependent on whether students explored, the discussion on constructive tasks is confined mainly to exploration with the software boxes. As in the previous sections, the problem is used as the analytical unit for comparing the constructive tasks and approaches.

As noted in Section 5.5.1 (p.139), students all scored differently for the constructive task depending on the problem, with students scoring the highest in Problem 2's constructive task followed by Problem 1 and Problem 3's constructive task (Finding 5 in Annex 5, p.326). Unlike the interpretive tasks, there was no significant difference between the performance for the higher and lower mathematics confidence students in the constructive tasks (Finding 7 in Annex 5, p.326). Further, as noted in Section 5.6.2 (p.150), the students who explored the constructive tasks were significantly more likely to obtain a mark (Finding 14 in Annex 7, p.329) and also that exploration was dependent on software box and mathematics confidence (Finding 13 in Annex 6, p.328). Hence, any discussion of the scores obtained by students has to be taken within this context.

6.6.1 Problem 1: Performance, Approaches and Software Boxes

Starting with Problem 1, only 9 students were able to obtain a score for this constructive task. This task required the student to explain how the toy manufacturer should change his production if the profit on the toy train was increased by £1.The increasing of the profit by £1 should not make a difference to the production as the demand for toy trains (40 toy trains) had already been met and hence production should not change but the profit will increase. This answer was only true for students using the black-box and glass-box software. The students using the open-box software may have also obtained an alternative solution where demand was not met (see Appendix 6, p.313).

Four students obtained a score without using any software box, two each from the lower and higher mathematics confidence group. Seven students used the software boxes, of whom five were able to obtain an answer that yielded a score. Six of the students who did explore were from the higher mathematics confidence group. The two students from the lower mathematics confidence grouping (Participants 12 and 21) who did not use the software seemed to give an answer based on a guess rather than any real understanding of what was happening (Table 42).

Although Participant 21 seemed to be on the right track, her mention of the price being too high showed there was a lack of understanding. The reasons they provided for why the number of toys produced would remain constant pointed to a feeling rather than an explanation determined from the task, and they used this feeling to understand the economics of the problem. On the other hand, the students from the higher mathematics confidence grouping who did not explore showed some ability of understanding what would occur to the profit but then also reverted to using their own idea of economics. For example, Participant 21 suggested that the company would be pleased with the profits and would not change production. The remaining students (who did not explore their answers) tended to use their own economics' beliefs in explaining their answers and this perhaps influenced their mainly real-life explanations.

Table 42: Responses for Problem 1's constructive task by the lower and higher mathematics confidence students not using a software box for solving the task

Lower Mathematics Confidence
If the profit per train has increased by one dollar the number of trains being manufactured should by no means also increased if the farmer insists on keeping his profit at 1 dollar more. However, dependant on how much of a profit the farmer is operating by, perhaps, it may not be to his detriment to produce a couple more of these toy pieces. Conversely, if he is operating at a 'dead- beat' loss, it may be better for him to count his blessings and maintain production as is.
(Participant 12, F, BB, Lower MC = 2)
The same constraints apply there the total number of toys produced will be the same but the profit will be dependant on the consumers because if the price of is too high the toys will not be purchased
(Participant 21, F, GB, Lower MC = 4)
Higher Mathematics Confidence
So if the profit per train increases by 1 then the max profit would reach 140, I would say that production on trains should increase or something
(Participant 3, M, BB, Higher MC = 9)
If the increase in profit per train is \$ 1 then this would mean that the company receives an overall profit of 20 x \$1 = \$20 with the current constraints. This may mean that: *1)The company is pleased with the profits and decides not to increase production of the trains, or* *2)The company can reinvest in the trains and increase their train production...to further increase profits.*
(Participant 30, M, OB, Higher MC = 8)

It was previously observed (p.205) that some students confused the terms variables, coefficients and constraints and it persisted during the constructive task, particularly for those students who had lower mathematics confidence. For example, Participant 33 thought that the demand constraint $x \leq 40$ had to be increased to find the profit. He increased this constraint to $x \leq 40 + 1$ rather than changing the coefficient in the objective function:

> 43:21: *"Well, let's see here"*, looks at the screen and moves mouse around the last iterations, *"there ought to be a value of cost assign to each of them, so, you'll know if*

you have 40 times a pound you'll have 40 pounds, and if you had one £20 times maybe £2 then you'll have £40, and then you can just add in £1 to the first one and then see what additional … well, how much it makes extra."

44:02: *"Working back from that, you'll probably want to put that into one of our constraints",* hovers along the whole *x* column of all the iterations.

44:16: Looks back to the paper: *"Umm … the C constraint … maybe … we want to change that and then find a different solution. So, if we change the … if we move it up to 41, then make 41 toy trains then we will be making less soldiers probably."*

44:59: Clicks for the answer sheet: *"Could be completely wrong but [mumbles]"*

45:06: Types in answer: *"Increase profit by £1 may chance constraint C to x<= 40 +1 and since x=40 was our previous answer this may mean it would now mean x increases and y decreases",* and says, *"don't know how to do the symbols",* when typing in the *x*<=40+1.

(Participant 33, M, OB, Lower MC = 6)

It was likely, because *x* was found to be 40, which meant the profit from *x* would have been £40, he saw *x* as money rather than the number of toys being produced. This suggested that he saw the increase of profit as related not to the coefficient but rather to a change in the variable.

Further, some students had difficulty in distinguishing between profit and cost and gave their answers or explanations in terms of the price of the product increasing rather than the profit (see Table 43). This seemed to happen for both lower and higher mathematics confidence students.

Table 43: Confusion between profit and cost for both lower and higher mathematics confidence students

Lower Mathematics Confidence
The number of trains and soldiers purchased would decrease due to an increase in price
(Participant 32, F, OB, Lower MC = 5)
If the profit per train increased, this means the price of the train increased, if the price of the train is higher than the price of the soldiers, consumers would more likely purchase the cheaper item
(Participant 20, F, GB, Lower MC = 5)
Higher Mathematics Confidence
It would decrease the number of toys being sold because if the profit increases it means that the price for the item has gone up.
(Participant 27, F, OB, Higher MC = 8)
more soldiers would be sold as this means that the price of trains would have increased and the company would now have to make more soldiers
(Participant 2, M, BB, Higher MC = 8)

Looking at the explorations, it was mostly the higher mathematics confidence students who explored (Finding 11 in Annex 6, p.328), which may be because the students recognised the connection between profit and changing the coefficient, since all but one were able to get the answer. Participant 11 (F, BB, Higher MC = 7) was unable to obtain an answer because she confused the concept of profit and cost and instead reduced the coefficient of x by 1 rather than increasing it. There was only one student (Participant 38, F, GB, Lower MC = 5) in the lower mathematics confidence group who recognised the solution and rewrote the objective function as $z = 3x + y$. Whilst she understood that the profit was connected to the coefficient, she felt prior to solving the constructive task that x should increase and y should decrease to maintain the profit of £100. Hence, when she used the glass-box software to modify the mechanical task and calculated the new answer, she still maintained what she thought earlier. Perhaps, in a situation like this, Participant 38 may have benefited from prompted self-explanations.

Further, out of all the students who explored, four used the black-box software, two used the glass-box software and one used the open-box software. This provides some indication that black-box software may be slightly better for exploration and possibly explains why students with the black-box software were scoring slightly higher in the constructive tasks (Finding 8 in Annex 5, p.326). Even though the students had been randomly assigned to the different boxes, there was a larger number of higher mathematics confidence students using the black-box and a larger number of lower mathematics confidence students using the glass-box, which may have influenced the results. Further, there was one student (Participant 34, M, OB, Higher MC = 7) using the open-box software who recognised that the objective function had to be changed but did not test this using his software box. It is difficult to ascertain whether he knew what to change in the equation; he commented as follows:

> 40:20: *"Are we dealing with only facts or are we … or can we interpret a reason",* I tell him "you choose how you wish to answer it". Looks at the instructional materials.
>
> 40:49: Looks up at the screen. *"Ohhh… Ok … So, if the profit is increased by £1, then the z would change, so therefore changing the x and y values ok".*
>
> Which he follows by typing this as his answer:
>
> *Profit has increased by 1 therefore changing the initial equation and so giving different values for the number of toy soldiers.*
>
> (Participant 34, M, OB, Higher MC = 7)

This raises the question whether the use of the open-box software detracted him from checking whether this was correct, since the open-box software required a longer time and perhaps more cognitive effort for producing the answer. Or probably he could have been just using a perception-based answer. (Perception-based answers are looked at more closely when examining Problem 2's constructive task in the next section.) On the other hand, Participant 13 (M, GB, Higher MC = 8) had already determined the answer but used the software box to confirm his answer. Probably he could readily confirm his answer with the software since he was using the glass-box software which required less cognitive effort and time:

> *If the profit per train is increased, it would likely be more profitable to produce more trains and fewer soldiers. However Constraint C puts an upper bound on the number of trains that can be produced -- a bound which has already been achieved. Hence it is not possible to produce more trains and the number of toy trains and toy soldiers produced would remain the same. This was confirmed by solving the modified problem.*
>
> Participant 13 (M, GB, Higher MC = 8)

Therefore, the difficulty of having poor algebra persisted and influenced the performance on the constructive tasks and also affected the students' ability to distinguish between profit and cost, which occurred across mathematics confidence levels. Exploration seemed to arise in students in the higher mathematics confidence group, particularly those assigned to the black-box software. This links to the work of Coupland (2004), in that higher mathematics confidence students are the ones with a cohesive conception of mathematics and thus they may be able to use the software boxes to promote their understanding. The black-box software seems to lend itself more easily to exploring, which may also be a factor in students' ability to do well on the constructive task. Students, who had previously had a misconception of the answer before exploring, were able to try and justify their calculated answers by using the equations provided.

6.6.2 Problem 2: Performance, Approaches and Software Boxes

Across the constructive tasks, students scored the highest in Problem 2 (Finding 5 in Annex 5, p.326). In this task, students were asked how the production should change for a furniture company if the carpentry hours were increased. As there was already a surplus of carpentry hours, the production should remain the same.

Only three students were able to obtain a mark for their answers without exploring, two in the lower mathematics confidence group and one in the higher mathematics confidence group. The 23 students who did explore with the software were all able to obtain a mark (Finding 15 in Annex 7, p.329). Examining the transcripts of the students who used the software showed that students who explored identified quickly (usually within about 30 seconds) that they needed to change the right-hand side of the carpentry constraint and resolve the mechanical task. For example:

> 39:13: *"The number of hours available for carpentry has increased to 60 how does this change..."*. Clicks ok and get rids of the answer sheet.
>
> 39:22: *"Right. Click the reset button"*. Clicks reset button and then clicks input problem. *"So, that will be carpentry/ day. Constraint A is changed from 48 to 60"*. Changes the RHS of A from 48 to 60, *"OK",* clicks ok. *"Iteration",* clicks iteration. Then clicks answer sheet.

(Participant 9, F, BB, Higher MC = 7)

> 1:02:08: Reads the instructional materials
>
> 1:02:18: *"Ok .. right, Let's rerun it and change the input problem. [Mumbles], let's make a note of it 140, 2 .."* Writes down the answer on paper (??)
>
> 1:02:35: He resets the problem and changes 48 to 60 on the input problem form for the first constraint

(Participant 33, M, OB, Lower MC = 6)

The remaining 15 students did not use the software box to answer this constructive task, either because they chose not to do so or because it simply did not occur to them to do so. For example:

> 1:22:56: Clicks Q3C in the answer form and then instructional materials. *"Last part. If the number of hours of carpentry per day is increased from 48 to 60 …",* continues reading out the question.
>
> 1:23:20: *"Right … obviously,"* looks at the screen, *"Obviously first of all the total production would be increased",* looks down.
>
> 1:23:27: *"So, firstly",* begins to type the answer, *"total production would be increased, overall production …",* continues typing, *"kinda obvious",* continues typing, *"would be increased and in particular",* stops typing looks at papers, *" … they would increase the*

number of stools produced", continues typing, *"... would increase the number of stools produced as the solution to the first question, the first part of the question."*

(Participant 16, M, GB, Higher MC = 7)

Participant 16 thought that the answer was obvious and began to think of his answer fairly soon after reading the task. It was interesting that some students decided to use the software boxes whilst others did not. Whilst it was thought that this might have been related to the students' confidence in computers or Excel, an examination of students' self-identified computers and Excel confidence scores found that these confidence scores were similar between those who explored and those who did not. It was possible that some students did not explore because they thought the answers in the constructive tasks should be solely perception-based and that this formed the basis on how they would answer their future tasks.

For example, Participant 5 (M, GB, Higher MC = 8) when answering Problem 2's interpretive task asked whether [34:16] "*the answer is mine ... like, uh, mine, why it wasn't possible right?*" which may have carried over in his perception of how he should answer the constructive task. Similarly, Participant 30 (M, OB, Higher MC = 8) made a similar statement when answering the interpretive task, in that he asked [1:04:11] "*If I just ... ummm ... speak to you about my perception*" and continued "*Oh, it is just totally my perception*".

There did not seem to be any obvious link to the students' processing levels, mathematics confidence or self-explanations to explain why students were acting in this manner. However, one might conjecture that, since the students easily obtained an answer for the interpretive task (that is, no chairs were produced), and also provided a relatively easy explanation (albeit in most cases a wrong answer, that the demand was low), that they may have proceeded to Problem 2's constructive task with more confidence. That is, the students probably felt that this interpretive task was easy, which minimised any anxiety, leaving them able to conjecture or decide how to proceed with Problem 2's constructive task quite easily either by using the software boxes or making a quick explanation. Perhaps, therefore the act of exploring with the software box in this constructive task was not directly related to the students' mathematics confidence but rather to their perception of what the answer should be. Those with the preconception that the answers had to be explained by themselves were less likely to use the software box whilst those who had neither of these preconceptions and who were propelled with some sense of mathematics confidence could directly see that they could use the software box to resolve the task.

An alternative explanation comes from an idea put forward by Trouche (2000), who suggested that there were particular strategies to using technology in solving problems, based on the extent that students used technology (in his case, calculators: Section 2.7, p.38). For example, the theorist used references, interpretation and analogy, the rationalist used paper and pen, inference and proof, whilst the tinkerer used the calculator, investigation and accumulation whilst solving problems. It may be that students who chose not to use the software boxes and who followed their perception were acting as theorists or the rationalists. From the think-aloud protocol, it was difficult to determine which profile these students fitted into. However, Participant 30 indicated that he preferred working with pen and paper than on a computer, which may have resulted in him not taking to exploring with the computer:

> 53:16: After being prompted: *"Ok, I don't know ... when I'm doing maths ... sometimes I don't really talk to myself ... but just use a whole set of scrap paper ... never really do maths on the computer before ... let's see ... [mumbles] ... good".*

There may be a completely different explanation as to why students seem to have explored more for Problem 2. This may depend on how they saw the linear programming equations. Upon examining the answers that students provided for

Problem 1's constructive task, most students appeared to believe that the production would only be affected if the constraints had changed (see Table 44).

Table 44: Student's conception of the linear programming problem from Problem 1's constructive task

[….] because the constraints are still the same only the variables have changed
(Participant 15, M, GB, Lower MC = 5)
[…] Toy soldiers would decrease because of the limited resources and constraints
(Participant 19, F, GB, Lower MC = 6)
The number of trains and soldiers being sold would not change as the constraints were based on carpentry and painting, not profit.
(Participant 28, M, OB, Higher MC = 7)

Hence when Problem 2's constructive task indicated that a change in the constraint had occurred, the students recognised that production might change and hence the linear programming task had to be run again with the changed constraint to determine what will be the new production. Sometimes, the students had an inaccurate perception of what the answer should be before they ran the numbers again through the software, but when they got the new answer, they examined the equations further to determine the reason why this was occurring:

> 39:22: *"Right. Click the reset button".* Clicks reset button and then clicks input problem. *"So, that will be carpentry/ day. Constraint A, is changed from 48 to 60".* Changes the RHS of A from 48 to 60, *"OK",* clicks OK. *"Iteration",* clicks iteration. Then clicks answer sheet.
>
> 40:01: *"And it is exactly the same …",* (laughs), *"Ok."* Begins to type, *"Increasing the number of hours available carpentry had not change the results"*
>
> 40:43: *"Uh … it must be the case, there must be something else. Is a main constraint? Let's see x t, t is 8",* perhaps referring to the LP answer, *"that'll be 16. 2 … so, Constraint C is already … and",* begins to type again, *"Because no more furniture can be made because … Constraint C is already at its maximum with x=2 and t=8 because this equals 16 and constraint C can't exceed 16".* Clicks Ok and gets rid of answer sheet. Final answer: *"increasing the number of hours available for carpentry did not change the results because no more furniture can be made because constraint c is*

already at its maximum with x=2 and t-8 because this equals 16 and constraint c can't exceed 16"

(P2C: Participant 9, F, BB, Higher MC = 7)

Participant 9 thus had a perception that the production would change, which was why she indicated some surprise at [40:01]. She began to think that there had to be a reason for this [40:43] and then re-examined the equations to determine the reason for there not being a change.

Thus, this section has indicated that Trouche's categories do not explain why a majority of the students chose to explore this constructive task rather than other constructive tasks. If this was a predisposed strategy, then there should be roughly equal amounts of exploration across all the tasks. Thus, there may be other influences on the answering of the tasks: in particular, the students' conception of what linear programming represented and their perception of what the task demanded of them. In the first instance (that is, students' conception of linear programming), their conception, that increasing the right hand side of the carpentry constraint would change the production, led them to explore in Problem 2's constructive task because they knew what they had to change. However, in Problem 1's constructive task, the students' conception of profit and cost did not immediately connect with the conception of profit being attached to the coefficients of the variables. Instead their perception (the second instance) of what they thought the task wanted took over: that is, they started using real-life heuristics to help to solve the tasks.

6.6.3 Problem 3: Performance, Approaches and Software Boxes

Across the three constructive tasks, students performed poorly in Problem 3 (Finding 5 in Annex 5, p.326). They were asked to determine the highest value that the variable *u* could become. The students should have been able find this answer either by examining the constraints or by testing values of *u* (using Constraint D) with the software boxes.

Whilst in the other constructive tasks some students were able to obtain a mark without using the software boxes, in this task only students who used the software boxes were able to determine the answer. Students who had lower mathematics confidence were more likely to give up earlier, whilst those with higher mathematic confidence were more likely to persist in understanding or finding a solution. Of all the students who were unable to obtain a mark on this constructive task, 6 students wrote "I don't know" which included Participant 15 who tested values for *u* using the glass-box software. Otherwise the most popular answer seemed to be that *u* would be infinity (Participants 2, 8, 35, 38). Some of these answers are presented in Table 45.

Other answers suggested that *u* should not be higher 100, and this was probably because Constraint B was 100 even though *u* was not included in this constraint. The four out of the five students who gave this answer were from the lower mathematics confidence group (Participants 21, 26, 31, 32) whilst the other student was from the higher mathematics confidence group (Participant 30). Another answer that was popular was given by five students (Participants 11, 17, 28, 34, 39), who suggested that *u* should not be higher than 105. Their reason for this was that *u* should not be higher than the profit (*z*) they found, which was £105. Interestingly, four of the students who gave this answer were from the higher mathematics confidence group. Possibly students with higher mathematics confidence who although gave wrong answers were still able to provide in-depth explanations as to what was occurring, whereas the lower mathematics confidence students just gave any answer. The lower mathematics confidence students tended to provide general explanations as to why things were occurring, whilst the higher mathematics confidence students gave more specific explanations.

Table 45: Examples of answers provided for Problem 3's constructive task

***u* = Infinity**
u can be made to a value that is a multiple of 90, such as, 180. Due to this fact, u can have an infinite value as long as the other variables in the equation are multiplied by the same factor.
(Participant 2, M, BB, Higher MC = 8)
The largest value should be infinity because of the number system
(Participant 38, F, GB, Lower MC = 5)
***u* =100**
The max value for u could be 100 due to the presumption that u is the slack value.
(Participant 21, F, GB, Lower MC = 4)
Largest u value would be 100 because the maximum constraint value is 100.
(Participant 30, M, OB, Higher MC = 8)
***u* =105**
So a value of u grater than 90 - 105 so the value of t = 0 so as to maintain a profit of 105
(Participant 34, M, OB, Higher MC = 7)
Maximum value for u would be 105 since it cannot exceed this value
(Participant 17, M, GB, Lower MC = 6)

Eight students explored using the software for this constructive task, a similar number to those who explored Problem 1's constructive task; however, for Problem 3's constructive task only three students were able to find the correct or partially correct answers. Of these 8 students, four used the black-box software, three used the glass-box and one used the open-box software. All of the students using the glass-box and open-box software were from the lower mathematics confidence group, whilst those assigned to the black-box software were all from the higher mathematics confidence group. Three out of the four black-box software students were able to obtain an answer for this task. It appeared that the reason for them obtaining an answer was that they were able to explore a wider range of numbers for *u*.

For example, for the black-box software, Participant 9 tested 200 and 1000, Participant 7 used three numbers (100, 200, 300) and Participant 1 tested numbers 200, 1000, -100, -200, -1000 after first exploring what occurred if a constraint was removed (see Figure 36). The remaining student, Participant 3, only tried one number (125) as he was confirming a calculation that he had made.

The one student (Participant 32) using the open-box software tested one number (91). For the three students using the glass-box software, Participant 22 tested two numbers (95 and 100), Participant 15 tried four numbers (100, 105, 120 and 170) and Participant 38 tried one number (100). All four students using the glass-box and open-box software made the wrong conjecture in the end. Participant 15 who tested the most numbers eventually gave his answer as being "*I don't know*". Students needed to test numbers above 200 to make the correct conjecture, which is why the three students using the black-box software got it correct.

What was interesting here was that only students using the black-box software tested a large range of numbers and hence found the correct answer. The students using the other software boxes seemed to limit their exploration to smaller numbers. In particular, the open-box software student only tested one number. His limit to testing only one number possibly was because of the longer time or cognitive effort required to test numbers.

48:34: She changes the u coefficient in the last row to 0 and the RHS from 90 to 0 and performs the iteration. 49:23: After being prompted to talk: "*Yeah I just had tried the putting back the equations without the [internet] constraint without the u and gave the answer as … the best value for it to be 200 but I'm not sure if that is the maximum value, I'm still thinking about it*". Proceeds to look at the papers she has and correspond with the screen.	Changes the constraint to find the highest value for u
51:24: Clicks ok to get rid of the answer sheet and changes the u coefficient in the last row to 1 and the RHS to 200 and do the iteration. 52:21: She changes the RHS of the last row to 1000 and do the iteration 52:31: She clicks input problem highlights the coefficient of u in the last row but then looks back at her papers 53:34: After being prompted to talk: "*I'm trying something with …um … the input problem to see what different values the last constraint will give me*." 53:49: Changes the RHS of the last row as -100 and the coefficient of the u as -1 (I had to tell that to put it as -1 rather than just a -)	Test constraints for 200 and 1000
54:27: She re-changes the RHS of the last row to -200 and clicks iteration. 54:59: She changes the RHS of the last row to -1000 and clicks iteration. 55:52: I explain to her that I am not certain whether this software works with negative RHS and she says ok. 56:11: Goes back to looking at her papers. 57:35: After being prompted: "*Well I'm looking at the constraints to see if 200 is the highest value it can get and looking at Constraints A and B*."	Test constraints for -100, -200 and -1000

Figure 36: Think-aloud transcript for Participant 1 (F, BB, Higher MC = 8) whilst doing the constructive task for Problem 3

Further, participants using the glass-box and open-box software who had lower mathematics confidence did not seem to be keen to explore further. This was seen particularly with Participant 15, who gave up because the software was not giving him an answer. Bandura (1986) explained that:

> Weak self-precepts of efficacy are easily negated by disconfirming experiences, whereas people who have a strong belief in their own competence will persevere in their coping efforts despite mounting difficulties … (p.396) … the stronger the perceived self-efficacy, the more likely are persons to select challenging tasks, the longer they persist at them, and the more likely they are to perform them successfully (p.397)

The ease with which students were able to explore with the black-box software both in Problem 1 and Problem 3 may indicate why there was a significant difference in exploration among the software boxes. These students were generally the higher mathematics confidence students. According to Coupland (2004), students with cohesive concepts are more likely to use the software effectively. However, in this study the software mode available to the students seemed to also influence how the students used it. In particular, the black-box software was used most effectively by the students with cohesive concepts rather than the glass-box or open-box software. For the lower mathematics confidence group the software mode did affect their frequency of exploration and it may be that these students need guided help with a teacher to achieve any useful learning from the software boxes.

As most of the higher mathematics confidence students were using the black-box, it seemed that they may be well equipped with the kind of software box that could work to their advantage. However, those students who had lower mathematics confidence were mainly assigned to the glass-box software, and this software box possibly confused them more and probably caused them to treat it as black-box software. The lower mathematics confidence students, however, if they saw steps with the mathematical terms, were able to ask themselves appropriate questions. These questions then prompted them to look for information or to self-explain what these mathematical terms meant and how they related to each other, as was seen by the students using the glass-box software.

Further, students using the software boxes were able to test their initial perceptions, which were usually based on real-life. If a new solution from the software boxes contradicted their initial thoughts, then the students dug deeper to find out what was occurring. Also, students' perceptions of what should happen and their conceptions of linear programming helped them to decide whether they should explore, for example in Problem 1, the relationship between coefficients and profit, in Problem 2, the changing of the right hand side of the carpentry constraints, and in Problem 3 that *u* was related to a constraint.

6.7 Discussion

This chapter investigated the overarching research question:

How do students' approaches to the three task types and their performance on these tasks depend on the software box they have access to?

To answer this question, the data and findings from Chapter 5 were used to answer why students' performance was affected by their approaches. Further, the developed analytical framework was used as a lens to try to understand and interpret the data. Students' typewritten answers to the tasks and 8 students' transcripts of their think-aloud session were further used to explore the influence of students' approaches and mathematics confidence on their performance on the different tasks.

This section answers the overarching question by drawing together discussions from Chapters 5 and 6.

6.7.1 Performance and Tasks

The quantitative data confirmed Galbraith and Haines (2000a) findings that students tended to do worse in constructive (0.40) than interpretive tasks (0.98) and also that students would be most successful at the mechanical tasks (Section 5.5.1, p.139, Finding 2 in Annex 5, p.326). However, being successful at the mechanical task was expected given the nature of the software boxes. Galbraith and Haines had their students solve mechanical and constructive tasks by hand. It was interesting that students' poor performance in the constructive tasks in this current study persisted even when the students had the option of using the software boxes (Section 5.5.2, p.144). Hence the results from this study showed that software did not seem to help students perform better in the constructive task over the interpretive task.

Galbraith and Haines suggested for constructive tasks that the "interaction of conceptual and procedural knowledge where procedures had to be introduced by the student" (p.13) was the least developed. Using software boxes, that were able to interact with the procedural steps or even show the procedural steps, did not enable the students to introduce their procedural knowledge to the constructive task. However, it appeared instead that the students' mathematics confidence when using the software boxes impacted on the linking between students' procedural and conceptual knowledge in the constructive task provided that the appropriate software was available.

Students with higher mathematics confidence who used the black-box software were more likely to introduce procedural knowledge to the conceptual part of the constructive task, as demonstrated by their high frequency of exploration (Section 5.6.1, p.147 and Finding 13 in Annex 6, p.328). The black-box software provided the best tool for these higher mathematics confidence students to explore, possibly because the black-box software required the least cognitive effort (Section 6.6.3, p.231) and ensured the least mental fatigue (Section 2.5.2, p.29). As suggested by Coupland (2004), students who adopted a deep processing level (usually the ones with the higher mathematics confidence) were more likely to use software in a meaningful way for their learning. Therefore, whilst mathematics confidence influenced whether a student explored, having the appropriate tool may facilitate the connection of procedural and conceptual knowledge. Thus, to extend Coupland's (2004) statement, students with a deep processing level can use software in a more meaningful way for their learning, but this will be dependent on the software box. Black-box software can be most meaningful in constructive types of learning tasks. The open-box software and to some extent the glass-box software can allow some students to engage in more meaningful learning in mechanical tasks as they are able to see mathematical terms and make sense of them (Section 6.5.3, p.214).

Further, students' conception of how a task works (for example, in Problem 2's constructive task, that changing the right-hand side of the constraint would yield a different result) possibly affects whether they could find a connection between the conceptual and procedural knowledge. This is also related to students' perception of how they should solve a task. If students thought that the answer had to come from their perception only, they neglected the use of software. Perhaps the level of cognitive effort required to use the software box decreased the extent to which the student would bother to use it (Section 5.6.1, p.147).

6.7.2 Performance and Software Boxes

Whether one software box promoted more conceptual understanding than the others is open to debate. The open-box software forced students to try and understand the steps or at least make a conjecture on which pivot variable to choose (Section 6.3.2, p.190) as was suggested in Section 2.6.3 (p.37). Whilst most higher mathematics confidence students persisted in trying to understand and test their conjectures, lower mathematics confidence students were more likely to give-up and resort to rule-of-thumb or means-end strategies to survive the mechanical task-solving process. In the glass-box software, some students initially worked towards understanding the steps, but there was a tendency to try and find the answer as quickly as possible; as a consequence, they began to use the glass-box software as a black-box software which may mean they were not gaining any additional intellectual profit from the steps (Section 6.3.1, p.182). However, it was possible that seeing the steps allowed students to be familiar with terms as they appeared.

Therefore, in both the glass-box and the open-box software, it may be better for all users but in particular lower mathematics confidence students, if students were either prompted for self-explanations or provided with scaffolding questions, since the lower mathematics confidence students would more likely adopt a surface processing level when using the software boxes with the intention of just getting by.

6.7.3 The Approaches, Boxes and Mechanical Tasks

The extent and nature of the explanations that students generated depended on the tasks. Students solving the mechanical task in the open-box software were more likely to try and explain or conjecture what was occurring. Students in the glass-box software did this to a lesser extent whilst for the students using the black-box software there was no clear indication whether the students wanted to see the steps or not. The black-box software group was the least likely to engage in explanation during the solution process. This was due to the nature of the software boxes rather than the nature of the students.

6.7.4 The Approaches, Boxes and Interpretive Tasks

For the interpretive tasks, there did not seem to be any influence of the software boxes except for the glass-box in an indirect manner for Problem 3. The conjecture was that for Problem 3, the students using the glass-box software were confused about linear programming terms. Thus, when asked about a linear programming term in Problem 3, they then returned to the instructional materials to clarify their thoughts which led to them self-explaining in order to understand the concept of constraints (Section 6.5.3, p.214).

The most important part of any conceptual task that allowed students to start spontaneous self-explaining appeared to be providing appropriate prompts in the written task, in particular, the prompt or cue of ‘why?’. Asking students ‘what?’ did not appear to encourage self-explaining, whereas the use of ‘why?’ made the students think and possibly look for better explanations.

6.7.5 The Approaches, Boxes and Constructive Tasks

Students were found to explore more with the black-box software than the glass-box and open-box software for the constructive tasks (Section 5.6.1, p.147 and Finding 12 in Annex 6, p.328). This was possibly because the students required less cognitive effort when using the black-box software as suggested in Section 2.5.2 (p.29).

The self-explanations in the constructive tasks were more pronounced when the students were prompted with the ‘why?’ part of the task and was noted that this prompt also worked in the interpretive tasks (Section 6.5.3, p.214). The only influence that the software boxes had was when students used the software box and obtained the correct answer. At that point, students were able to self-explain regarding why the answer had occurred, but this appeared to be mostly relevant for students who had higher mathematics confidence and who used the black-box software. The ‘why?’ prompt also allowed students to provide explanations that were broadly based on real-life or mathematical explanations. Since the answers to the constructive tasks were mostly obtained through the use of mathematical principles, students who latched onto the mathematical explanations were more likely to get the constructive tasks correct (Finding 29 in Annex 9, p.332).

Students tended to use the real-life explanations when they could not connect their procedural with conceptual knowledge and, as mentioned previously, tended to answer based on their perceptions (Section 6.6.1, p.220). Thus, these students thought that a perception-based answer would be one that made sense of the materials within their context of knowledge. However, some students saw their context of knowledge as being from the social context rather than from the mathematical context. Hence, it may be that these students consider mathematical explanations as being un-natural in their sense-making mechanism.

It may be necessary when creating tasks, that task designers provide 'why?' cues to encourage students to self-explain. Further, whilst the glass-box and the open-box software may be useful in understanding the procedural algorithm, providing some kind of prompting mechanism to help students self-explain or provide feedback could encourage students engaging with the linear programming in a deep way and possibly help the students to make connections with the underlying linear programming concepts. Further, the black-box software seemed to be mostly appropriate for the higher mathematics confidence students. To that extent, it can be recommended for such students.

6.8 Concluding Remarks

This chapter began with a discussion on the type of qualitative data collected from the students (Section 6.2, p.175). Further using the quantitative results from Chapter 5, qualitative links between the performance scores (although not for the mechanical task), the three approaches and the software boxes were investigated for the mechanical (Section 6.3), interpretive (Sections 6.4 and 6.5) and constructive tasks (Section 6.6).

The results from Chapters 5 and 6 have shown that performance is dependent on two approaches, the explorations and the explanations. Students who explored with the software boxes in the constructive tasks were more likely to obtain the correct answer and, having achieved the correct answer, were better able to explain why there were possible relationships between variables and constraints in the linear programming problem (Sections 5.6 and 6.6). Students who provided mathematical explanations were also more likely to have a high performance score. Whilst there was no conclusive evidence that students with a deep processing level had higher performance scores, it appeared that most students with higher mathematics confidence provided better explanations, possibly as a result of their deep processing of the information.

Chapter 7. Conclusion

"Doubt, indulged and cherished, is in danger of becoming denial; but if honest, and bent on thorough investigation, it may soon lead to full establishment of the truth."
- Ambrose Bierce

7.1 Introduction

Sections 5.9 (p.165) and 6.7 (p.237) discussed how the data from this thesis answered the research questions outlined in Section 2.10.1 (p.50). This chapter begins with discussing the main contributions of the research (Section 7.2, p.244). This is followed by a reflection on the research process with recommendations for research and practice (Section 7.3, p.258) with a discussion on the limitations of the research (Section 7.4, p.261). The implications of these findings for teachers, software developers and students are also discussed (Section 7.5, p.265). The chapter concludes with suggestions for future research (Section,7.6 p.268).

7.2 Main Contributions of the Research

Four main contributions are identified, namely the characterisation of the software boxes, the development of the remote observation process, the development of an analytical framework and the findings with respect to the software boxes and tasks.

7.2.1 The Software Boxes

Prior to this research, there were limited investigations into how students perform and their preferred approaches when using the three software boxes. The research by Horton *et al.* (2004) had only compared two kinds of software box: black-box and glass-box. At the start of this research, there were no comparisons of the open-box software with any other of the software-boxes and, so far has been determined, there is no current study of this kind except for this one. Prior to this research, all that was known when comparing all three of the software boxes was that one showed no steps, one showed steps and one provided interactivity at steps. Now, through this study, there is evidence relating to a comparison of the three software boxes in terms of student's performance, the software boxes' computation capacity and the approaches that students used (see Table 46).

Table 46: Comparison of the three software boxes based on eight characteristics

Characteristics	Black-Box	Glass-Open	Open-Box
Feature	No Steps	Shows Steps	Interacts at Steps
Time for computing tasks	Fast: one click of the button (less than 5 seconds)	Average: several clicks of the button (less than 10 seconds)	Slow: Need to go through several iterations before computing an answer (at least 45 seconds)
Procedural Knowledge Learnt	None	Depends on whether they are willing to look at the steps	Force to find some understanding of what to do – even if it is just to learn a heuristic
Student Performance	Interpretive: ✓✓ Constructive: ✓✓	Interpretive: ✓✓✓ Constructive: ✓	Interpretive: ✓✓✓ Constructive: ✓
Exploration (constructive tasks)	High: ✓✓✓	Average: ✓✓	Low: ✓
Exploration	High: ✓✓✓ Low: ✓	High: ✓ Low: ✓✓	High: ✓ Low: ✓
Predisposed to Explanations	Real-Life: ✓✓ Maths: ✓✓✓	Real-Life: ✓✓✓ Maths: ✓✓	Real-Life: ✓ Maths: ✓✓✓
Deep/Surface Processing Levels	Undetermined/ same	Undetermined/ same	Undetermined/ same

Further, the study highlighted the challenges of finding mathematical software that was capable of providing all three software modes. The researcher programmed the Excel sheets to be representative of the three software boxes, and this helped highlight the particular challenges in deciding how many and which steps should be included in both the glass-box and open-box software.

The main outcome of this research is that there is no clear indication that one software box is best for learning in all contexts. For example, as seen in Section 5.6.1 (p.147), the black-box software seemed to encourage more exploration by students in constructive tasks (44% exploration) than in any other of the software boxes. Both the glass-box and open-box software appeared to detract this behaviour in students (33% and 22% respectively). However, by using the open-box software the students were often encouraged to try to develop their own understanding of how to proceed in solving mechanical tasks (not necessarily the correct understanding) (Section 6.3.2, p.190). Further, the open-box software appears to keep students in a more mathematical frame of mind, perhaps because of its high ratio of mathematical to real-life explanations (1.71) compared to the other software boxes (Section 5.7.2, p.156). The glass-box software appears to be the in-between software: intermediate in terms of encouraging exploration and intermediate in terms of allowing students to understand the steps involved in solving a mechanical task.

7.2.2 Remote Observation

The second contribution of this research was the development of the remote observation method for observing students interacting with software. The equipment and software requirements for a remote observation study are presented in Table 47 as determined in this study.

Table 47: Equipment and software required by the researcher and participant for the remote observation process

Remote Observation Equipment/Software	Researcher	Participant	Example
Webcam	✓	✓	
Microphone	✓	✓	
Speakers/ headset	✓	✓	
Application sharing software	✓	✓	e.g. Windows Messenger, Unyte Application Sharing
Voice/video conversation software	✓	✓	e.g. Windows Messenger, Skype
Broadband (or better) internet connection	✓	✓	
Screen capture software	✓		e.g. Camtasia Studio
Audio wave in and out recording software	✓		e.g. vEmotion
Large computer RAM	✓		At least 1GB
Large Hard-drive	✓		At least 1GB available per participant

There are many integrated software packages that are able to application share, but this was one of the first studies that used application sharing together with web cameras for researching understanding by students. During the development of the remote observation method, another colleague was investigating the use of Netviewer, an application-sharing integrated package in which web cameras and the think-aloud protocol were also used to observe how participants used course materials available on the OpenLearn website (see San Diego and McAndrew, 2007).

A comparison of Netviewer and the two modes of remote observation (with Windows Messenger and Skype) used in this study is presented in Table 48. The main difference between this study's remote observation method and using Netviewer, besides the cost, is that this study's remote observation method allowed synchronous conversation whilst in Netviewer, voice-conversation worked in a two-way radio transmitter mode. The two-way radio transmitter mode means that when one person (either the researcher or participant) wants to talk they have to click a button. A think-aloud session in Netviewer will thus mean that after prompting students to talk, the student will be required to click a button before beginning to talk and thus there is a possibility of two problems occurring:

1. The student may 'forget' to click the button to talk and thus their self-explanations may be lost until reminded to click the button
2. Clicking the button to talk provides a constant reminder to the participants that a researcher is observing them and may increase any Hawthorne effect (Section 4.3.3, p.109).

The current remote observation method developed does not have these problems to the same extent since students are free to spontaneously think-aloud without being reminded they have to click a button to talk.

Table 48: Comparison of Windows Messenger, Skype and Netviewer for remote observation

Property	Windows Messenger with Netmeeting	Skype with Unyte	NetViewer
Web camera and headphones	Yes	Yes	Yes
Integrated package	No – several windows	No – several windows	Yes – one window
No. of computers	1 or 2	1 or 2	1
Software required for observing students	Windows Messenger and Netmeeting (Application Sharing)	Skype software, Unyte Application Sharing and web browser	Web-browser and Netviewer (for researcher)
Software participants installs	Before session: Windows Messenger During Session: Netmeeting	Before session: Skype During session: Load links into a web-browser	During session: Netviewer (for participants) which loads onto the web-browser
Synchronous voice/video conversation	Yes	Yes	No: Walkie-Talkie Mode
Chat facility	Yes	Yes	Yes
Point and guide	No – unless control is undertaken by the researcher	No – unless control is undertaken by the researcher	Yes
Video synchronisation	Difficult if two computers used, with one computer simpler	Difficult if two computers are used, with one computer simpler	Easy – only records one video
Cost	Cheap	Cheap	Expensive

A protocol for the remote observation study was created for those students who were not recruited via a gatekeeper. This protocol is presented in Table 49.

Table 49: Protocol for remote observation when contacting the participant

Protocol for Remote Observation	How/Why
1. Make contact with the participants and obtain email address	Use either a web-forum where they can contact you via email or have a list of participants' emails where you can contact them directly. Ensure that the participant has the equipment/ software required.
2. Send email detailing the research, a web-link for the consent form and background questionnaire (if using)	Set up a web-form in which you can collect the data. Ensure web-form has fields such as name and email address to identify the participant.
3. Send email indicating receipt of consent form	To confirm that the participant has agreed to the remote observation exercise and not someone else. The assumption is that the participants are only able to access their email.
4. Send email which has a) date and time for the participant to sign in for the remote observation b) researcher's contact number and request a contact number and c) any instructions/ materials required by the participant	a) The date and time indicates when the participant should enter the session. The researcher should sign on a few minutes before the session time to welcome the participant. b) In the event of internet/computer failure either the participant or the researcher should be able to contact each other through the phone. c) Any instructional materials that the participants need to print or read in advance should also be sent. Alternatively, materials can be set up online as in a web-page and the participant can be redirected to these either before or during the remote-observation session through a web-link.
5. At the beginning of the remote observation session, ask for consent of audio and video recording, either through instant messaging or voice conversation	Asking the participants consent again to audio/video record them minimizes the problems for not having the signed consent forms.

For students recruited through a gatekeeper where a mini-lab is set up, this protocol is changed slightly. Step 1 is different as the gatekeeper recruits the students and liases with the students to arrive at the appropriate time. The gatekeeper then informs the researcher of the session time. Step 2 occurs during the session and Steps 3 and 4 are omitted completely with the gatekeeper providing the instructional materials to the students during the session.

Through investigation of the web camera angles, the best position of web cameras for observing the students during the study was identified (see Figure 37, p.252). A web camera angle that provides a good view of the desk and the student ensures that the researcher can see when the students are reading, writing, using the mouse or typing. However, as some web cameras are integrated into laptops this means that the angle is only of the face of the student. From these integrated web cameras, it is still possible to get an idea whether the students are reading or paying attention to the screen by looking at the students' head and eye movements (look-up or down) combined with their use of mouse or keyboard. In a situation where the web camera does not provide a good angle, it is helpful to the researcher if the student talks about their actions during the think-aloud session. The think-aloud data stating the students' actions can then be triangulated with the video data to determine what the students are doing.

Good use of web camera	Useable
	
Able to see the desktop, the monitor and can follow when he is reading on the screen or looking at paper	No view of the monitor, but able to see when picking up papers to read or when moving right hand with mouse

Figure 37: The web cameras angles in remote observation

7.2.3 Implications of Remote Observation

Using the remote observation method for collecting data opens a number of avenues. Firstly, researchers are no longer restricted to collecting data from particular geographic regions because of the logistics. Through remote observation almost any population can be accessed albeit any population on the internet with the required computer equipment. Secondly, if gatekeepers can be organised in the locality, then they can set up the remote observation mini-laboratories where the necessary computer equipment can be provided. This means that participants who have had to travel a long distance to a user-laboratory now may only have to travel to their nearest local mini-laboratory centre. Thirdly, home-bound participants such as the disabled can take part in studies without the necessary travel and upheaval required in traditional user-lab situations.

There are some considerations that have to be taken into account about the remote observation method presented. This study involved mostly young people who were accustomed to the internet; perhaps older participants who are unfamiliar with the

Internet may not like the impersonality of a remote observation setting. Also, broadband internet is essential for remote observation; otherwise application sharing can become a slow and frustrating process for both the researcher and the participant because of slow processing speeds. Thus, using participants from regions or countries (particularly developing countries) where broadband is not widely available will be almost impossible. Finally, unlike in a user-lab situation, the researcher has no control over outside influences such as mobile phones ringing, friends interrupting the remote observation session or knowing whether the participants are surfing the internet during the session. In all cases in this study where there were mobile calls, the participants asked permission from the researcher to answer the mobile; they then proceeded to indicate that they were in a research session and cut the conversation short. Thus, participants can probably police themselves.

7.2.4 Development and Extension of the Analytical Framework

In Chapter 2, an analytical framework was developed to aid in the analysing of the data (Figure 3, p.48). A number of educational and learning perspectives could have been used to explain the variations in students' performance with the software boxes including Vygotsky's instrumental method (Vygotsky, 1930/1997), activity theory (Leontiev, 1947/1978), distributed cognition (Hutchins, 1995) and instrumental genesis (Vérillon and Rabardel, 1995). However, the decision was made to focus on the students' understanding of the tasks rather than their use of the software box (since they did not always use the software boxes). Thus these theories that were primarily focused on the software tool such as the instrumental method, activity theory, distributed cognition and instrumental genesis were not considered.

Cognitive load theory (Sweller, 1988; Sweller and Chandler, 1991) for a time was a serious contender for explaining students' understanding. However, when it became clear that there were diminished think-aloud data during the tasks, it had to be excluded, even though the initial quantitative analysis had shown that students were exploring more with black-box software (the software with the least cognitive load). Thus other theories that influenced students' understanding were considered such as those relating to conceptual/procedural knowledge, processing levels, self-explanations and self-confidence.

The literature review had shown that self-confidence, self-explanations and processing levels all impacted on performance and were hence incorporated in the experimental design. It was only when further investigation of the literature showed that processing levels were related to self-confidence that a conceptual map was drawn about the possible relationships between the identified approaches and self-efficacy. The conceptual map started crudely but was eventually formed into the triangular format shown in Figure 3 (p.48). Using the analytical framework, gaps in the literature were identified. From these identified gaps, the association between self-confidence and processing levels for mathematics students was confirmed through data collected in Supporting Study 1.

The framework incorporated two of the approaches (explanations and processing levels) and self-confidence for showing how these may influence performance. However, the framework at that time did not incorporate the approach of exploration, as the relationship of exploration to performance was uncertain. From Section 6.6 (p.220), students who explored the constructive task and obtained an answer were more likely to provide the correct explanations as to why a change resulted. This however was dependent on two factors: confidence level and the software box. Students with high mathematics confidence using the black-box software were more likely to explore the constructive tasks (56%) compared to students using the other software boxes whilst those with low mathematics confidence were more likely to explore using the glass-box software (41%) than with any other of the software boxes (Section 5.6.1, p.147). In either case, when students arrived at a solution through the explorations, this presented them with an opportunity to reflect on the reason for that particular solution. Through this reflection, students were at times able to provide the correct explanations and hence have higher performance scores (see particularly Section 6.6.2, p.226). This relationship is presented in Figure 38. Further, in Chapter 6, it was shown qualitatively that students who had higher mathematics confidence provided better quality or more detailed explanations (not necessarily correct) than those students with lower mathematics confidence (see particularly Section 6.4.1, p.198). Thus, a tentative relationship between mathematics confidence and quality of explanations is established and is included in the updated framework in Figure 38.

Finally, through Supporting Study 1 (Section 4.2, p.95), the relationship between mathematics confidence and deep mathematical processing level was established through factor analysis for mathematics students and is updated accordingly in this framework. This is the first evidence of a relationship between mathematics confidence and the deep processing level. This is consistent with previous suggestions in the literature that academic self-confidence is related to the deep processing level (for example Duff, 2004). Although this finding does not directly relate to the research questions set out in this thesis, it does however add to the approaches to study and self-efficacy literature in the mathematical domain.

Whilst a relationship between quality or quantity of explanations and the processing levels may exist, there was no clear empirical evidence to show such a relationship, and this remains to be ascertained.

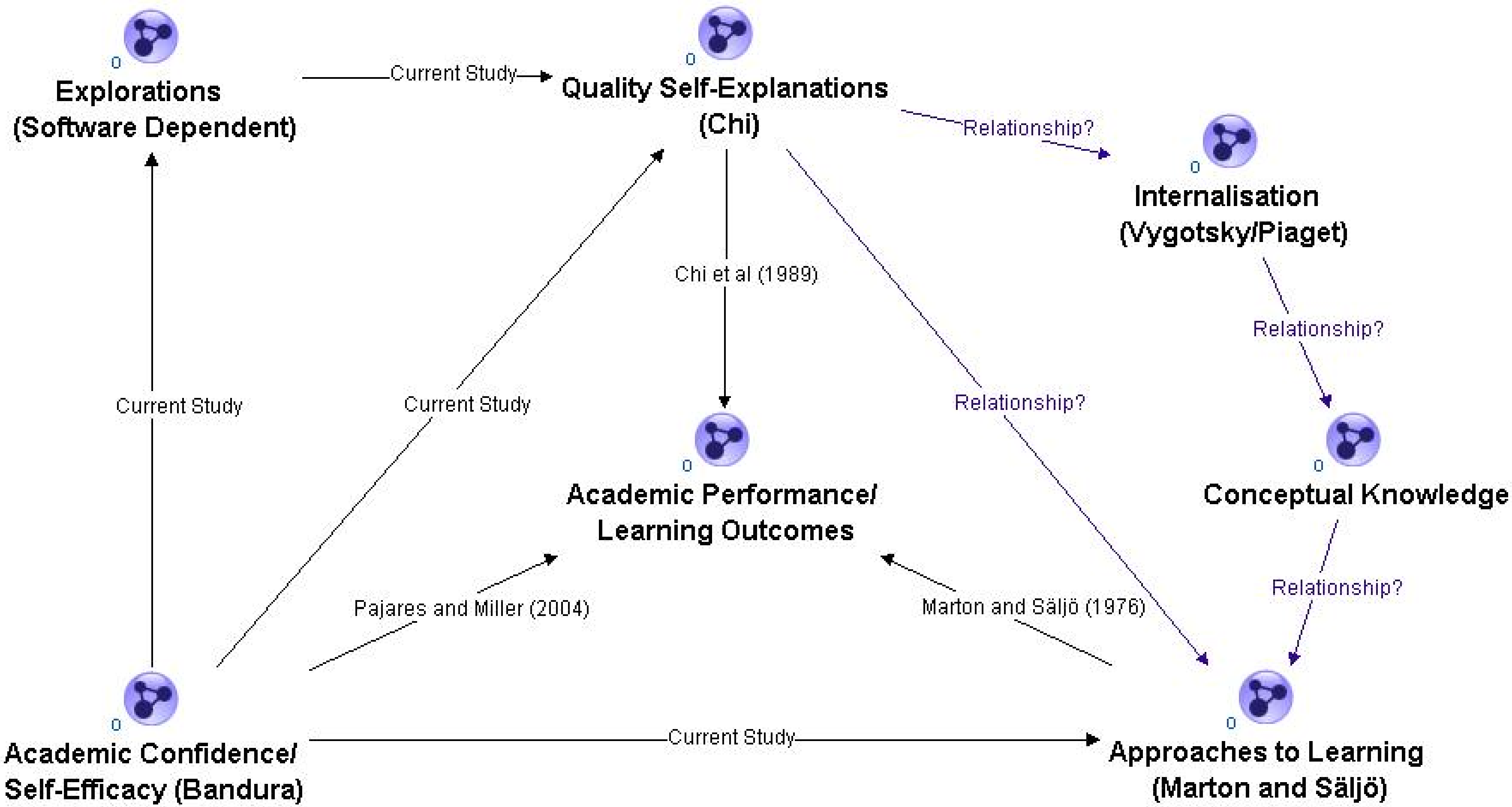


Figure 38: Updated analytical framework for approaches and performance in mathematical understanding

7.2.5 Conceptual and Procedural Knowledge with Software Boxes

The last contribution to be discussed concerns the tasks. All three tasks based on Galbraith and Haines (2000a) required differing levels of conceptual and procedural knowledge for their solution. Previously, Galbraith and Haines had shown that there were differences in scores obtained between the mechanical, interpretive and constructive tasks for polynomial functions. The difference in scores was confirmed in this study with students scoring higher in the interpretive (0.98) than the constructive tasks (0.40) but this time in the linear programming domain (Section 5.5.1, p.139). Mechanical tasks were not counted in this study because all calculations were completed through the computer. The tasks developed by Galbraith and Haines were not used in an environment where students were allowed to choose between using the computer or pen-and-paper. Previously Leinbach, Pountney and Etchells (2002) had developed all three task types for polynomial and trigonometric functions; however, all tasks had to be completed via a computer algebra system (CAS). Leinbach *et al.* did not report on the scores obtained by their students. Thus, this research adds to this literature some evidence that students' conceptual-procedural link (shown through the constructive task) is still poor even when there is a software package available to aid in the procedural calculations.

Also, whilst this study did not investigate whether mathematical software promoted conceptual knowledge more than not using software, as found by Heid (1988) (and to a lesser extent O'Callaghan (1998) and Palmiter (1991), see Section 2.3.3, p.21), this research shows no strong evidence that one software box promotes conceptual knowledge more than another. However, this research does provide evidence that for higher mathematics confidence students the ease of computations can possibly lead to better use of their conceptual knowledge. This is evidenced through higher mathematics confidence students doing better on the constructive tasks with the black-box software than any of the software boxes (Section 5.5.2, p.144).

7.3 Reflection on the Research Process

There were several lessons learnt during this research which are outlined under separate headings below. These lessons cover from the start (e.g. gaining permissions and approvals) to the end (e.g. data analysis) of the research process.

7.3.1 Permissions and Approval

For Supporting Study 1 (Section 4.2, p.95), the permissions for carrying out an online survey with Open University students were approved through its Student Research Project Panel (SRPP). One of the conditions for the approval was that the researcher needed permission from each of the course managers before sending out the online surveys. Most online surveys of students from the Open University are carried out through the Survey Office. The Survey Office is responsible for inviting the students through an email link to log-in and answer the questionnaire, rather than it being the responsibility of the course manager, lecturer or the researcher.

Some course managers readily gave their approval for their students to take part in the research, particularly for courses in technology and mathematics. However in the psychology courses, the course managers were initially reluctant to provide any approval as their main concern was that this research had no relevance to their students. After several email negotiations over a few weeks, approval was received. Perhaps the more important lesson learnt is that the negotiating process can be a prolonged one where initial setbacks can be overcome with persistence through providing the necessary arguments and supporting material needed to allay the concerns of the interested parties.

This experience further highlights the necessity of planning well in advance for receiving any type of permissions. The informal communication between the researcher and the interested parties was the key to obtaining consent, and may be the wisest initial course of action before applying for the approval.

7.3.2 Testing of Tasks and Software

The two pilot studies in this thesis involved the testing of the software boxes and the three task types by Galbraith and Haines (2000a) concerning expected values and linear programming. Pre-testing the tasks was essential. For example, in Pilot Study 2 (Expected Values) some students chose not to use the software boxes because the tasks could be easily solved through pen-and-paper methods. It was from this Pilot Study that the methods began to change with respect to the choice of the mathematical domain, as it was now apparent that students would use the software boxes only:

- if the researcher asked them to
- if the task was too complex to solve by hand or
- if they saw a need to use it.

Therefore the lessons learnt from the pilot studies were that one should always test the tasks to determine:

- whether students understood the mathematical domain from the instructional materials (both expected values and linear programming)
- whether the chosen tasks could provide the required data for answering the research questions (for example, students engaging with the tasks to produce think-aloud data).

For both pilot studies, the software boxes were programmed by the researcher in MS Excel using Visual Basic for Applications (VBA). As mentioned before, this software is provided in the attached CD. It was necessary for the researcher to create and program the software boxes as no mathematical software could be identified to represent all three software boxes. It was helpful to develop a schema with regard to how the software boxes should operate when solving linear programming problems (see Figure 13, p.114). This was critical when deciding whether the open-box software (represented by OB in the figure) should have two interaction levels (steps 1 to 9) or only one (Steps 1 to 4 and 7 to 9).

It was important to pilot the software boxes to know whether a) students could use the software boxes, b) the programmed software modes were representing the black-box, glass-box and open-box principles and c) there were any errors or glitches in the software. Through testing, it was observed that students were able to use the software boxes and that there were no obvious user-computer interaction challenges. Although the developed software interface was sufficient for the research, making a friendlier interface (Holzinger, 2005) might enhance the usage of the software boxes.

7.3.3 Recruiting Participants via Facebook and Gatekeepers

As noted in Section 3.6.1, (p.82), students for the Main Study were initially recruited via popular social networking sites such as Facebook. Whilst posts were made in the Facebook forums as well as paid electronic fliers at this website, the recruitment of students was poor (3 students responded from which only 1 participated).

This suggests that there are limitations of recruiting students via social-networking websites and perhaps using participants who are enthusiasts may gain a higher response rate (Section 3.6.1, p.82). Further, minimising the equipment requirements of participants may also increase the response rate.

The use of gatekeepers in this research was advantageous in gaining access to students especially where the gatekeepers were able to take responsibility for acquiring and setting up the equipment.

7.3.4 Statistical Analysis

An initial statistical analysis of the Main Study (Chapter 5, p.124) used a number of analyses of variance (ANOVAs) and regressions for analysing the different approaches. These however proved complex to understand and to follow an argument. There were added problems in that the statistical analysis had heterogeneity of variance and unequal sample sizes, which all impacted on the robustness of the ANOVAs. Changing the statistics to using frequencies and non-parametric statistics (such as chi-squares) provided not only a different and interesting way of looking of the data but a simpler way of presenting it as well.

7.4 Scope and Limitations

This research focused on university students learning linear programming as isolated individuals using one software box. The findings about the software boxes should extend to other mathematical domains provided that the solving of the task is sufficiently complex (unlike that of expected values, as shown in Pilot Study 2). Extending these findings is only suitable if any developed software boxes show procedural steps and the tasks given to students are either interpretive or constructive.

Further, students in secondary schools may well exhibit similar behaviour (performance and approaches) to university students when using the software boxes, especially where they are encouraged to learn more independently such as during GCE A-Levels. However, further research should be conducted using secondary-school students and in other mathematical domains to show that these findings are generalisable.

Whilst the research focused on learning at the individual level, it does recognise that other types of learning such as collaborative learning may influence how students learn with the different modes of software. Also, the sample used in the research was mainly from Trinidad and Tobago. A study by Watkins and Biggs (2001) suggested that cultural background influenced students' levels of processing, especially in Chinese students. It is however unclear whether the observed influence was due to the culture or the learning context as these were confounded with each other in this study. Trinidad and Tobago's secondary-school education and teaching systems are modelled on the British system, and their university system is also partly modelled on the British system (and uses the same degree awards) (Section 3.6, p.81). Therefore, these results may be representative of students in Western educational systems.

The students' performance was only measured in two out of the three tasks (the interpretive and constructive tasks, not the mechanical task). From Pilot Study 2, the students felt that the tasks, particularly the mechanical and interpretive tasks, were all the same (Section 4.3.3, p.109). They expressed some sense of frustration in having to repeat the same procedures. To circumvent this issue, for Pilot Study 3 and the Main Study, all three task types were designed around a problem to minimise a feeling of sameness. Through this, the interpretive and constructive tasks were based on students calculating the mechanical task correctly. Getting the mechanical task correct was not a difficult accomplishment, as the students only had to ensure that they had inputted the correct numbers. Some students inputted numbers incorrectly, but, because the interpretive and constructive tasks were dependent on the calculated solution, the researcher prompted them to ensure that they inputted the numbers correctly. Thus, the mechanical task was ignored when investigating performance. Of course, in real learning situations, a researcher will not be present to ensure the correction of these mistakes, and so performance on the mechanical task will vary because of student error.

The university where this research was carried out was responsible for additional limitations. The study was conducted at The Open University, which is primarily an online and distance-education university. Unlike in other studies where students on a campus were recruited, finding participants presented a challenge. However, this gave rise to the development of remote observation, a new method for observing students at a distance. This meant that students at other campuses or elsewhere were now able to take part in the study without the need for long-distance travelling. The sample size of 38 used in this study was primarily achieved by asking university gatekeepers to recruit students in their departments at another university. This sample size provided a trade-off between good statistical power and depth of analysis of multiple-stream qualitative data (Section 3.6, p.81).

One of the issues that arose during the statistical analysis of the data was that the Mathematics Confidence variable which was used as a covariate was causing a violation of one of the underlying assumptions of the Analysis of Covariance (ANCOVA). This underlying assumption was the homogeneity of slopes. This violation suggested that Mathematics Confidence was influencing the interaction effect of software Boxes and Tasks. That is, the performance scores for Box by Tasks were being influenced by different levels of Mathematics Confidence (such as an increase or decrease in Mathematics Confidence). However, the ANCOVA statistical test could not be used to test how the different levels of Mathematics Confidence were influencing performance scores as it violated one of the underlying assumptions. Therefore, to circumvent this issue, the Mathematics Confidence variable was dichotomised using its median, a common practice in the psychology domain. Students were then assigned into lower and higher Mathematics Confidence groupings. From this grouping, differences in scores between lower and higher Mathematics Confidence group of students could be found.

However, one has to note, as with most statistical analysis, the differences in the group scores are based on pooled data. Thus, whilst a few students with lower mathematics confidence scored highly, on average the lower mathematics confidence group scored statistically lower. Taking this into account, the qualitative examples of students' approaches and answers used in this thesis were therefore frequently from an average scoring student in these two mathematics confidence groupings. In some instances, particularly when discussing students' answers, both low and high scoring answers within both mathematics confidence groupings were used to illustrate how students from the two groups differed.

Further, as the students used in this sample were mainly from natural sciences, engineering and medical disciplines, it meant that the mathematics confidence displayed by the students were generally quite high. There may be a larger disparity in students' performance and quality of explanations if students from other disciplines such as Arts had been considered. The reason for this is because Art students' mathematics confidence might be even lower than the lower mathematics confidence students used in this study. Thus, the statistical data which pointed to differences in performance due to mathematics confidence may become more apparent when using students with even lower mathematics confidence (e.g. Arts students).

The experiment conducted in this study used one-to-one observations, rather than just post-test data. The remote observation method provided a large corpus of data, not all of which could be fully analysed. The data collected included over 76 hours in video, about 45 pages of hand-written observation notes and over 11,000 words in typewritten answers. Of the 38 students, eight students' sessions were analysed in full, which is similar to the number of transcripts used by Chi *et al.* (1989) in their investigation of self-explanations (10). The eight students were chosen as they were representative both of the software boxes and of mathematics confidence. Their transcripts were used for illustrating the quantitative data and to investigate further any anomalies noted from this data. Thus, these students' transcripts were not used in isolation but instead served as the first port of call to check trends. If a trend was observed in this data, then it was cross-checked with the observation notes of all the participants and then cross-checked with the video data. For example, this triangulation of data was used in testing why low mathematics confidence students using the glass-box software achieved higher marks in Problem 3's interpretive task (Section 6.5.3, p.214).

7.5 Implications of Main Study

This section discusses the implications of the results of the Main Study for software designers and educators with particular interest on how this may affect students' understanding of a mathematical topic.

7.5.1 Implications for Software Designers

As was noted from Section 7.2.1 (p.244), there was no software box that was best for all circumstances. Thus it may mean that software designers should ideally allow all mathematical software to have the option to operate in the three software box modes. This would make certain that teachers and students can choose the appropriate software box for their teaching and learning objectives respectively. Just as Buchberger (1990) had recommended (Section 1.3, p.3), students may use the glass-box and open-box for grasping the procedural knowledge. Through doing this, the students will not only become familiar with the terms involved in solving problems (such as was seen with the glass-box students who tried to understand the term basic variable) but may also be able to think mathematically, that is, with mathematical symbols (as was seen with students using mainly the open-box software). When students have become accustomed to the terminologies and procedural steps, they probably can then move on to using the black-box software for solving problems and applying their conceptual knowledge. However, it is possible that higher mathematics confidence students may be able to make this transition quicker.

Further, as found in Chapter 6, whilst the glass-box and the open-box software were useful by some for understanding the procedural algorithm, it was clear that students were not engaging thoroughly with the steps to gain any conceptual knowledge (for example, lack of trying to understand the slack values). Thus, if the software modes of glass-box and open-box software can include a prompting mechanism to help students self-explain what the steps mean, it may help students to build their conceptual knowledge and encourage students to engage with the mathematical topic in a deep way. Through the engagement of the mathematical topic, students will then be able to make connections with the underlying mathematical concepts. One has to guard against the likelihood that students may make the wrong self-explanations. Thus, it may be useful to encourage students to first self-explain via a prompt and then let the software box provide an explanation from which the students can build their conceptual knowledge. However, one challenge of such a software design is to ensure that students do not skip over the prompt and go directly to the explanation as they would lose the valuable process of making conceptual connections.

Perhaps to ensure that students make explanations, teachers should prompt the students instead of the software or at least provide some type of scaffolding questions. Through the prompts and the scaffolding questions, students can engage with the steps in a deeper way from which they can understand the reasoning behind the calculations rather than just the arithmetic or algorithmic process.

7.5.2 Implications for Educators

Based on the results of how students used the software boxes, lecturers or teachers should not use only one software box but a mixture of the software boxes to teach any topic, depending on the objectives of the class. If the objective is to learn procedural knowledge, then the open-box software may be more appropriate. Perhaps, as Winston (1996) suggested, learning the procedural steps may occur more often in the mathematical disciplines and hence these students may be able to make more use of the open-box software (see Section 1.3, p.3).

With guided learning, the glass-box software may help students to understand the steps before proceeding to solving the task on their own, such as by using the open-box software. If the objective of the class is to explore different solutions, then using black-box software may be more appropriate as it allows students to quickly acquire answers and perhaps see connections between procedural and conceptual knowledge (as evidenced in students being able to explain why changes were occurring in the constructive tasks).

As noted in Chapter 6, students were able to self-explain more when the tasks were provided with ‘why’ prompts rather than ‘what’ prompts. Thus teachers may want to consider creating tasks that ask students ‘why’ in order that students are encouraged to make self-explanations and tap into their conceptual knowledge. Teachers may consider using real-life application problems, as students seemed to engage with these problems more. These problems should however be realistic representations of the world as was the case with the linear programming problems. However, this raises an issue as realistic problems may cause students to use real-life heuristics and thus provide answers based on what their perception of the case should be rather than using any mathematical grounded knowledge.

Whilst in some problem types, students may be able to perform well regardless of their mathematics confidence level, it appears for other problems that some students with low mathematics confidence may be at a disadvantage. Building the mathematics confidence of a student is thus necessary as mathematics confidence impacts on the quality of self-explanations that the students make and hence the conceptual knowledge they are able to create.

Building mathematics confidence may not be an easy undertaking, as students' mathematics confidence may be deep-rooted such as being unable to perform correctly simple mathematical tasks such as algebra. As was observed in this study, even at the tertiary level, students still have poor algebra skills and may be a reflection on how algebra is taught at the secondary school level. These poor algebra skills generally point to poor conceptual knowledge (for example, some students were unable to distinguish between a variable and a coefficient). Teaching algebra which promotes both conceptual and procedural knowledge is thus needed if students are to apply these skills with confidence and appropriateness. The results from Chapter 6 showed that students using the glass-box and the open-box software with lower mathematics confidence were more likely to understand the concept of variables unlike those lower mathematics confidence students using the black-box, and it may mean the use of the glass-box and open-box software may help these weaker students build their conceptual knowledge whilst solving problems in a procedural manner.

7.6 Future Work

There are several areas that prompt additional research:

1. This current research involved students who assessed themselves as generally having a high mathematics confidence (Section 5.3.2, p.134). It would be useful to compare formally the differences in performance and approaches between equal group sizes of students with low and with high mathematics confidence
2. Remote observation was used mostly with young students, for whom a protocol was developed. Using remote observation with older

populations or people with disabilities could lead to remote observation protocols tailored to these populations

3. A more extensive investigation is needed to determine whether there is a relationship between self-explanations and the processing levels. Perhaps assessing students' explanations and processing levels as they proceed from one task to another could aid in this illumination.

4. This study only looked at students using the software boxes on their own. Perhaps investigating situations where students are collaborating can highlight any different approaches that groups of students adopt when using the software boxes.

5. At the time of the study, the remote observation method used MSN messenger and Skype because of its ability to have synchronous voice conversation and a separate video recording device. With recent software such as Elluminate which allows both synchronous voice conversation and ability to record sessions, the remote observation protocol should be investigated into how it can be modified to take advantage of this new technology.

6. Finally, this research concentrated on investigating software boxes in one mathematical domain. This research should be extended to see whether students undertake similar approaches and have similar performance in other mathematical domains when using the software boxes. Further this research could also be extended to cognate mathematical domains such as Physics or Engineering.

7.7 Concluding Remarks

This chapter outlined a number of achievements of this research. What started off as an investigation into whether students perform better in black-box, glass-box and open-box software has also provided a number of other interesting research outcomes, in particular the development of the remote observation method and the analytical framework.

Section 7.2 (p.244) discussed the main contributions of this research. In particular, this study has now addressed the questions of how students perform on the three task types for all three software boxes and how the software boxes influenced the approaches undertaken by the students. The implications of the use of the software boxes for teachers and software programmes were highlighted, particularly that all software modes are useful (the black-box software for exploring and the glass and open-box software for understanding steps). This section also provided a remote observation protocol that could apply in future studies. Finally, the analytical framework, which was developed in Section 2.9 (p.46), was modified to include all empirical contributions from this research, namely the links between self-confidence, explorations and self-explanations.

Section 7.3 (p. 258) provided a reflective piece on the research process in particular it highlighted the challenges of gaining permissions and approvals. Further this section reiterates the challenges associated with recruiting participants via the internet and the boons of access to appropriate gatekeepers.

The limitations of the research were also addressed (Section 7.4, p.261). Whilst the limitations in some cases were unavoidable such as the unavailability of undergraduate volunteers, this yielded new opportunities such as the development and testing of the remote observation method.

The implications of the Main Study for both educators and software designers were discussed in Section 7.5 (p.265), where recommendations for designing mathematical software were made as well as suggestions for when lecturers should use the software boxes.

To round off this chapter, other contexts for using the software boxes were suggested for future research, as well as other avenues where the remote observation method could be used, such as in developing countries or people with disabilities.

References

Albritton, M. D., McMullen, P. R. and Gardiner, L. R. (2003), "OR/MS content and visibility in AACSB-accredited US business programs", *Interfaces*, **33**(5), pp. 83-89

Aleven, V. A. W. M. M. and Koedinger, K. R. (2002), "An effective metacognitive strategy: learning by doing and explaining with a computer-based cognitive tutor", *Cognitive Science*, **26**(2), pp. 147-179

Anderson, J. R. (1995), *Cognitive Psychology and Its Implications*, 4th Ed., W.H. Freeman and Company, New York

Anstey, K. J., Salim, A., Lord, S. R., Hennessy, M., Mitchell, P., Mill, K. and Von Sanden, C. (2007), "Our correct use of ANCOVA yields acceptable results", *Journal of the International Neuropsychological Society*, **13**(2), pp. 371

Association of Commonwealth Universities (2008), *Commonwealth universities yearbook*, 82nd Ed., Association of Commonwealth Universities, London

Atkinson, R. K., Renkl, A. and Merrill, M. M. (2003), "Transitioning from studying examples to solving problems: effects of self-explanation prompts and fading worked-out steps", *Journal of Educational Psychology*, **95**(4), pp. 774-783

Bache, R. and Mullerberg, M. (1990), "Measures of testability as a basis for quality assurance", *Software Engineering Journal*, **5**(2), pp. 86-92

Bandura, A. (1977), "Self-efficacy: toward a unifying theory of behavioral change", *Psychological Review*, **84**(2), pp. 191-215

Bandura, A. (1986), *Social Foundations of Thought and Action*, Prentice-Hall, Inc., New Jersey

Biggs, J., Kember, D. and Leung, D. Y. P. (2001), "The revised two-factor study process questionnaire: R-SPQ-2F", *British Journal of Educational Psychology*, **71**(1), pp. 133-149

Biggs, J. B. (1987), *Student Approaches to Learning and Studying*, Australian Council for Educational Research, Hawthorn, Victoria

Bloom, B. S. (1956), *Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain*, David McKay Co Inc., New York

Bloom, B. S., Hastings, J. T. and Madaus, G. F. (1971), *Handbook on the Formative and Summative Evaluation of Student Learning*, McGraw Hill, New York

Boaler, J. (1993), "The role of contexts in the mathematics classroom: Do they make mathematics more "real"?" *For The Learning of Mathematics*, **13**(2), pp. 12-17, In Hennessy, 1993

Buchberger, B. (1990), "Should students learn integration rules?" *ACM SIGSAM Bulletin*, **24**(1), pp. 10-17

Buchberger, B. (2002), "Computer algebra: the end of mathematics?" *ACM SIGSAM Bulletin*, **36**(1), pp. 16-19, Translated by C. Schutzky

Campbell, D. T. and Stanley, J. C. (1963),"Experimental and quasi-experimental designs for research and teaching", In: N. L. Gage (Ed.), *Handbook of Research on Teaching*, Rand McNally, Chicago, IL, pp. 171-246

Chi, M. T. H. (2000),"Self-explaining expository texts: The dual processes of generating inferences and repairing mental models", In: R. Glaser (Ed.), *Advances in Instructional Psychology*, Lawrence Erlbaum Associates, Mahwah, NJ, pp. 161-238

Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P. and Glaser, R. (1989), "Self-explanations: how students study and use examples in learning to solve problems", *Cognitive Science*, **13**(2), pp. 145-182

Chi, M. T. H., De Leeuw, N., Chiu, M.-H. and Lavancher, C. (1994), "Eliciting self-explanations improves understanding", *Cognitive Science*, **18**(3), pp. 439-477

Clough, G. M. (2005), *Mobile Devices in Informal Learning*, MSc Dissertation, Institute of Educational Technology, Open University, Milton Keynes, UK

Cohen, J. (1983), "The cost of dichotimization", *Applied Psychological Measurement*, **7**(3), pp. 249-253

Cohen, J. (1988), *Statistical Power Analysis for the Behavioral Sciences*, Erlbaum, Hillsdale, NJ

Coupland, M. (2004), *Learning with New Tools*, PhD Thesis, Department of Information Science, University of Wollongong, Wollongong

Crawford, K., Gordon, S., Nicholas, J. and Prosser, M. (1998a), "Qualitatively different experiences of learning mathematics at university", *Learning and Instruction*, **8**(5), pp. 455–468

Crawford, K., Gordon, S., Nicholas, J. and Prosser, M. (1998b), "University mathematics students' conceptions of mathematics", *Studies in Higher Education*, **23**(1), pp. 87-94

Creswell, J. W. (2003), *Research Design - Quantitative, Qualitative and Mixed Methods Approaches*, Sage Publications, Thousand Oaks, CA

Cretchley, P., Harman, C., Ellerton, N. and Fogarty, G. (2000), "MATLAB in early undergraduate mathematics: an investigation into the effects of scientific software on learning", *Mathematics Education Research Journal*, **12**(3), pp. 219-233

Cretchley, P. C. and Galbraith, P. (2002), "Mathematics or computers? Confidence or motivation? How do these relate to achievement?" *Proceedings of the 2nd International Conference on the Teaching of Mathematics at the Undergraduate Level*, Hersonissos, Crete, John Wiley and Sons Inc., http://www.math.uoc.gr/~ictm2/Proceedings/pap318.pdf

Crowe, D. and Zand, H. (2000), "Computers and undergraduate mathematics 1: setting the scene", *Computers and Education*, **35**(2), pp. 95-121

Crowe, D. and Zand, H. (2001), "Computers and undergraduate mathematics 2: on the desktop", *Computers and Education*, **37**(3-4), pp. 317-344

D'Alonzo, K. T. (2004), "The Johnson-Neyman procedure as an alternative to ANCOVA", *Western Journal of Nursing Research*, **26**(7), pp. 804-812

Dana-Picard, T. and Steiner, J. (2004), "The importance of 'low-level' CAS commands in teaching engineering mathematics", *European Journal of Engineering Education*, **29**(1), pp. 139-146

de Jong, T. and Ferguson-Hessler, M. G. M. (1996), "Types and qualities of knowledge", *Educational Psychologist*, **31**(2), pp. 105-113

Delaney, H. D. and Maxwell, S. E. (1983), "ANCOVA and repeated measures: dealing with heterogeneity of regression", *67th Annual Meeting of the American Educational Research Association*, Monteal, Quebec

Dreyfus, T. (1994),"The role of cognitive tools in mathematics education", In: R. Biehler, Scholz, R. W., Sträßer, R. and Winkelmann, B. (Eds.), *Didactics of Mathematics as a Scientific Discipline*, Kluwer, Dordrecht, pp. 117-120

Drijvers, P. (2000), "Students encountering obstacles using CAS", *International Journal of Computers for Mathematical Learning*, **5**(3), pp. 189-209

Duff, A. (2004), "The revised approaches to studying inventory (RASI) and its use in management education", *Active Learning in Higher Education*, **5**(1), pp. 56-72

Dwyer, C., Hiltz, S. R. and Passerini, K. (2007), "Trust and privacy concern within social networking sites: A comparison of Facebook and MySpace", *Proceedings of the Thirteenth Americas Conference on Information Systems*, Keystone, Colorado, http://csis.pace.edu/~dwyer/research/DwyerAMCIS2007.pdf

Edmunds, R. and Richardson, J. T. E. (2009), "Conceptions of learning, approaches to studying and personal development in UK higher education", *British Journal of Educational Psychology*, **79**(2), pp. 295-309

Encyclopædia Britannica (2009),"Linear programming", In: *Encyclopædia Britannica Online*, Last accessed: 27th April, 2009: http://www.britannica.com/EBchecked/topic/342203/linear-programming

Ericsson, K. A. and Simon, H. A. (1984), *Protocol Analysis: Verbal Reports as Data*, MIT Press, Cambridge, MA

Ericsson, K. A. and Simon, H. A. (1998), "How to study thinking in everyday life: contrasting think-aloud protocols with descriptions and explanations of thinking", *Mind, Culture and Activity*, **5**(3), pp. 178-186

Fennema, E. and Sherman, J. A. (1976), "Fennema-Sherman mathematics attitudes scales: instruments designed to measure attitudes toward the learning of mathematics by females and males", *Journal for Research in Mathematics Education*, **7**(5), pp. 324-326

Field, A. (2000), *Discovering Statistics Using SPSS for Windows*, Sage Publications Inc., London, UK

Flavell, J. H. (1976),"Metacognitive aspects of problem solving", In: L. B. Resnick (Ed.), *The Nature of Intelligence*, Erlbaum, Hillsdale, NJ,

Galbraith, P. and Haines, C. (2000a), "Conceptual mis(understandings) of beginning undergraduates", *International Journal of Mathematical Education in Science and Technology*, **31**(5), pp. 651-678

Galbraith, P. and Haines, C. (2000b), *Mathematics-Computing Attitude Scales*, Monographs in Continuing Education, City University, London

Gass, S. I., Hirshfeld, D. S. and Wasil, E. A. (2000), "Model world: the spreadsheeting of OR/MS", *Interfaces*, **30**(5), pp. 72-81

Gentner, D. R. (1988),"Expertise in typewriting", In: M. T. H. Chi, Glaser, R. and Farr, M. J. (Eds.), *The Nature of Expertise*, Lawrence Erlbaum Associates Inc., Hillsdale, NJ, pp. 1-21

Gibbs, G., Habeshaw, S. and Habeshaw, T. (1988), *53 Interesting Ways to Appraise Your Teaching*, Technical and Educational Services, Bristol

Gilmore, G. C. (2007), "Inappropriate use of covariate analysis renders meaningless results", *Journal of the International Neuropsychological Society*, **13**(2), pp. 370

Gist, M. E. and Mitchell, T. R. (1992), "Self-efficacy: a theoretical analysis of its determinants and malleability", *Academy of Management Review*, **17**(2), pp. 183-211

Glass, G. V., Peckham, P. D. and Sanders, J. R. (1972), "Consequences of failure to meet assumptions underlying the fixed effects analyses of variance and covariance", *Review of Educational Research*, **42**(3), pp. 237-288

Große, C. S. and Renkl, A. (2006), "Effects of multiple solution methods in mathematics learning", *Learning and Instruction*, **16**(2), pp. 122-138

Hammersley, M. and Atkinson, P. (1995), *Ethnography - Principles in Practice*, 2nd Edition, Routledge, London

Hausmann, R. G. M. and Chi, M. T. H. (2002), "Can a computer interface support self-explaining?" *International Journal of Cognitive Technology*, **7**(1), pp. 4-14

Heid, M. K. (1988), "Resequencing skills and concepts in applied calculus using the computer as a tool", *Journal for Research in Mathematics Education*, **19**(1), pp. 3-25

Heid, M. K. and Edwards, M. T. (2001), "Computer algebra systems: revolution or retrofit for today's mathematics classrooms?" *Theory Into Practice*, **40**(2), pp. 128-136

Hennessy, S. (1999), "The potential of portable technologies for supporting graphing investigations (Abstract)", *British Journal of Educational Technology*, **30**(1), pp. 57-60, Last Accessed: 26th Jan, 2009: http://www.educ.cam.ac.uk/people/staff/hennessy/BJET_Lit_review.pdf (Full document)

Hiebert, J. and Lefevre, P. (1986),"Conceptual and procedural knowledge in mathematics: an introductory analysis", In: J. Hiebert (Ed.), *Conceptual and Procedural Knowledge: The Case of Mathematics*, Lawrence Erlbaum Associates Inc., Hillside, NJ, pp. 1-27

Holzinger, A. (2005), "Usability engineering methods for software developers", *Communications of the ACM*, **48**(1), pp. 71-74

Horton, R. M., Storm, J. and Leonard, W. H. (2004), "The graphing calculator as an aid to teaching algebra", *Contemporary Issues in Technology and Teacher Education*, **4**(2), pp. 152-162

Hosein, A. (2005), *Teaching of Linear Programming: Variation Across Disciplines and Countries*, MSc Dissertation, Institute of Educational Technology, Open University, Milton Keynes

Howell, D. C. (2002), *Statistical Methods for Psychology*, 5th Edition, Duxbury Thomson Learning, Pacific Grove, CA

Howell, J., Miller, P., Park, H. H., Sattler, D., Schack, T., Spery, E., Widhalm, S. and Palmquist, M. (2005), *Reliability and Validity*, Writing@CSU, http://writing.colostate.edu/guides/research/relval/ [Retrieved 15th Nov, 2008]

Hutchins, E. (1995), *Cognition in the Wild*, MIT Press, Cambridge, MA

Jones, R. M. and Fleischman, E. S. (2001), "Cascade explains and informs the utility of fading examples to problems", *23rd Annual Conference of the Cognitive Science Society*, Edinburgh, Scotland

Kadijevic, Đ. and Krnjaic, Z. (2003), "Is cognitive style related to link between procedural and conceptual mathematical knowledge?" *The Teaching of Mathematics*, **6**(2), pp. 91-95

Kadijevich, D. and Haapasalo, L. (2001), "Linking procedural and conceptual mathematical knowledge through CAL", *Journal of Computer Assisted Learning*, **17**(2), pp. 156-165

Kember, D. (1997), "A reconceptualisation of the research into university academics' conceptions of teaching", *Learning and Instruction*, **7**(3), pp. 255-275

Kendal, M. and Stacey, K. (1999), "CAS, calculus and classrooms", O. Zaslavsky (Ed.), *23rd Conference of the International Group for the Psychology of Mathematics Education*, **3**, Haifa, Israel pp. 129-136

Kendal, M. and Stacey, K. (2001), "The impact of teacher privileging on learning differentiation with technology", *International Journal of Computers for Mathematical Learning*, **6**(2), pp. 143-165

Landsberger, H. (1958), *Hawthorne Revisited*, Cornell University Press, Ithaca, NY

Laurillard, D. (1979), "The processes of student learning", *Higher Education*, **8**(4), pp. 395-409

Leinbach, C., Pountney, D. C. and Etchells, T. (2002), "Appropriate use of a CAS in the teaching and learning of mathematics", *International Journal of Mathematical Education in Science and Technology*, **33**(1), pp. 1-14

Lenth, R. V. (2006), "Java Applets for Power and Sample Size [Computer software]", http://www.cs.uiowa.edu/~rlenth/Power/, Last Retrieved 4th March, 2008.

Leontiev, A. N. (1947/1978), *Activity, Consciousness, and Personality*, Prentice Hall, Inc., Englewood Cliffs, NJ

Lindsley, D. H., Brass, D. J. and Thomas, J. B. (1995), "Efficacy-performance spirals: a multilevel perspective", *Academy of Management Review*, **20**(3), pp. 645-678

Marton, F. (1975), "On non-verbatim learning: 1. Level of processing and level of outcome", *Scandinavian Journal of Psychology*, **16**(1), pp. 273-279

Marton, F. and Säljö, R. (1976), "On qualitative differences in learning I. Outcome and process", *British Journal of Educational Psychology*, **46**(1), pp. 4-11

Maxwell, S. E. and Delaney, H. D. (1993), "Bivariate median splits and spurious statistical significance", *Psychological Bulletin*, **113**(1), pp. 181-190

Meagher, M. (2000), *Curriculum and assessment in an age of computer algebra systems*, Digest, Report No.: EDO-SE-00-04, ERIC Clearinghouse for Science, Mathematics, and Environmental Education

Myers, G. J. (1979), *The Art of Software Testing*, John Wiley & Sons,

O'Callaghan, B. R. (1998), "Computer-intensive algebra and students' conceptual knowledge of functions", *Journal for Research in Mathematics Education*, **29**(1), pp. 21-40

Olejnik, S. and Algina, J. (2003), "Generalized eta and omega squared statistics: measures of effect size for some common research designs", *Psychological Methods*, **8**(4), pp. 434-447

Owen, S. V. and Froman, R. D. (2005), "Why carve up your continuous data?" *Research in Nursing & Health*, **28**(6), pp. 496-503

Pajares, F. and Miller, M. D. (1994), "Role of self-efficacy and self-concept beliefs in mathematical problem solving: a path analysis", *Journal of Educational Psychology*, **86**(2), pp. 193-203

Palmiter, J. R. (1991), "Effects of computer algebra systems on concept and skill acquisition in calculus", *Journal for Research in Mathematics Education*, **22**(2), pp. 151-156

Piaget, J. (1972), *The Principles of Genetic Epistemology*, Routledge, London

Pierce, R. and Stacey, K. (2001), "Observations on students' responses to learning in a CAS environment", *Mathematics Education Research Journal*, **13**(1), pp. 28-46

Pierce, R., Stacey, K. and Barkatsas, A. (2005), "A scale for monitoring students' attitudes to learning mathematics with technology", *Computers and Education*, **48**(2), pp. 285-300

Plake, B. S., Ansorge, C. J., Parker, C. S. and Lowry, S. R. (1982), "Effects of item arrangement, knowledge of arrangement test anxiety and sex on test performance", *Journal of Educational Measurement*, **19**(1), pp. 49-57

Prosser, M. and Trigwell, K. (1999), *Understanding Learning and Teaching*, Open University Press, Buckingham

Ramanau, R. (2007), *Perceptions of Transactional Distance: A Comparative Study of Distance Learning in the UK and Russia*, PhD, Institute of Educational Technology, The Open University, Milton Keynes

Ramsden, P. and Entwistle, N. (1981), "Effects of academic departments on students' approaches to studying", *British Journal of Educational Psychology*, **51**, pp. 368-383

Renkl, A. (1997), "Learning from worked-out examples: A study on individual differences", *Cognitive Science*, **21**(1), pp. 1-29

Renkl, A. and Atkinson, R. K. (2003), "Structuring the transition from example study to problem solving in cognitive skill acquisition: a cognitive load perspective", *Educational Psychologist*, **38**(1), pp. 15-22

Renkl, A., Atkinson, R. K. and Große, C. S. (2004), "How fading worked solution steps works – a cognitive load perspective", *Instructional Science*, **32**(1-2), pp. 59-82

Richardson, J. T. E. (2000), *Researching Student Learning: Approaches to Studying in Campus-Based and Distance Education*, Open University Press, Buckingham

Richardson, J. T. E. (2005), "Students' perceptions of academic quality and approaches to studying in distance education", *British Educational Research Journal*, **31**(1), pp. 7-27

Richardson, J. T. E. and Edmunds, R. (2007), *A Cognitive-Developmental Model of University Learning*, SOMUL Working Paper, Report No.: 4, The Open University

Rittle-Johnson, B. and Alibali, M. W. (1999), "Conceptual and procedural knowledge of mathematics: does one lead to the other?" *Journal of Educational Psychology*, **91**(1), pp. 175-189

Rittle-Johnson, B., Siegler, R. S. and Alibali, M. W. (2001), "Developing conceptual understanding and procedural skill in mathematics: an iterative process." *Journal of Educational Psychology*, **93**(2), pp. 346-362

Roy, M. and Chi, M. T. H. (2005),"The self-explanation principle in multimedia learning", In: R. E. Mayer (Ed.), *The Cambridge Handbook of Multimedia Learning*, Cambridge University Press, New York, pp. 271-286

Ruthven, K., Hennessy, S. and Brindley, S. (2004), "Teacher representations of the successful use of computer-based tools and resources in secondary-school English, mathematics and science", *Teaching and Teacher Education*, **20**(3), pp. 259-275

San Diego, J. P. and McAndrew, P. (2007), "Learning from ‘OpenLearner-interactions’ using digital research techniques", *OpenLearn 2007*, Milton Keynes pp. 95-98

Sapsford, R. (1999), *Survey Research*, Sage Publications Inc., London

Savenye, W. C. and Robinson, R. S. (1996),"Qualitative research issues and methods: an introduction for educational technologists", In: D. H. Jonassen (Ed.), *Handbook of Research for Educational Communications and Technology*, Simon and Schuster Macmillan, New York, NY, pp. 1171-1195

Schultz, B. B. (1985), "Levene's test for relative variation", *Systematic Zoology*, **34**(4), pp. 449-456

Schunk, D. H. (1991), "Self-efficacy and academic motivation", *Educational Psychologist*, **26**(3&4), pp. 207-231

Schworm, S. and Renkl, A. (2006), "Computer-supported example-based learning: when instructional explanations reduce self-explanations", *Computers and Education*, **46**(4), pp. 426-445

Shrout, P. E. and Fleiss, J. L. (1979), "Intraclass correlations: uses in assessing rater reliability", *Psychological Bulletin*, **86**(2), pp. 420-428

Skemp, R. R. (1976), "Relational understanding and instrumental understanding", *Mathematics Teaching*, **77**(1), pp. 20-26

Smagorinsky, P. (1998), "Thinking and speech and protocol analysis", *Mind, Culture and Activity*, **5**(3), pp. 157-177

Smith, G. H., Wood, L. N., Crawford, K., Coupland, M., Ball, G. and Stephenson, B. (1996), "Constructing mathematical examinations to assess a range of knowledge and skills", *International Journal of Mathematical Education in Science and Technology*, **27**(1), pp. 65-77

Smith, K. B. (1994), "Studying different methods of technology integration for teaching problem solving with systems of equations and inequalities and linear programming", *Journal of Computers in Mathematics and Science Teaching*, **13**(4), pp. 465-479

SPSS-Knowledgebase (2008), *Resolution 22133: Repeated Measures with Constant Covariates in GLM*, http://support.spss.com/Student/Studentdefault.asp

Star, J. R. and Seifert, C. (2006), "The development of flexibility in equation solving", *Contemporary Educational Psychology*, **31**(3), pp. 280-300

Strickland, P. and Al-Jumeily, D. (1999), "A computer algebra system for improving student's manipulation skills in algebra", *International Journal of Computer Algebra in Mathematics Education*, **6**(1), pp. 17-24

Sweller, J. (1988), "Cognitive load during problem solving: effects on learning", *Cognitive Science*, **12**(2), pp. 257-285

Sweller, J. and Chandler, P. (1991), "Evidence for cognitive load", *Cognition and Instruction*, **8**(4), pp. 351-362

Tabachnick, B. G. and Fidell, L. S. (2007), *Using Multivariate Statistics*, 5th Edition, Pearson Education Inc., Boston

Tall, D. (1994),"Computer environments for the learning of mathematics", In: R. Biehler, Scholz, R. W., Strasser, R. and Winkelmann, B. (Eds.), *Didactics of Mathematics as a Scientific Discipline*, Kluwer, Dordrecht, pp. 189-199, Google Books

Tall, D., Gray, E., Bin Ali, M., Crowley, L., DeMarois, P., McGowen, M., Pitta, D., Pinto, M., Thomas, M. and Yusof, Y. (2001), "Symbols and the bifurcation between procedural and conceptual thinking", *Canadian Journal of Science, Mathematics, & Technology Education*, **1**(1), pp. 81-104

Towle, N. J. and Merrill, P. F. (1975), "Effects of anxiety type and item-difficulty sequencing on mathematics test performance", *Journal of Educational Measurement*, **12**(4), pp. 241-249

Trejo, L. J., Knuth, K., Prado, R., Rosipal, R., Kubitz, K., Kochavi, R., Matthews, B. and Zhang, Y. (2007),"EEG-based estimation of mental fatigue: convergent evidence for a three-state model", In: *Foundations of Augmented Cognition*, Springer, Berlin, pp. 201-211

Trouche, L. (2000), "La parabole du gaucher et de la casserole à bec verseur: ètude des processus d'apprentissage dans un environnement de calculatrices symboliques", *Educational Studies in Mathematics*, **41**(3), pp. 239-264

Tudge, J. R. H. and Winterhoff, P. A. (1999),"Vygotsky, Piaget, and Bandura: perspectives on the relations between the social world and cognitive development", In: P. Lloyd and Fernyhough, C. (Eds.), *Lev Vygotsky: Critical Assessments*, Routledge, London, pp. 311-338

Vale, C. M. and Leder, G. C. (2004), "Student views of computer-based mathematics in the middle years: does gender make a difference?" *Educational Studies in Mathematics*, **56**(2-3), pp. 287-312

Van Breukelen, G. J. P. and Van Dijk, K. R. A. (2007), "Use of covariates in randomized controlled trials", *Journal of the International Neuropsychological Society*, **13**(5), pp. 903-904

Vérillon, P. and Rabardel, P. (1995), "Cognition and artifacts: A contribution to the study of thought in relation to instrumented activity", *European Journal of Psychology of Educational Psychologist*, **10**(1), pp. 77-101

Vygotsky, L. S. (1930/1997),"The instrumental method in psychology", In: R. W. Rieber and Wollock, J. (Eds.), *The Collected Works of L.S. Vygotsky - Problems of the Theory and History of Psychology*, Plenum Press, New York, pp. 85-89

Vygotsky, L. S. (1931/1997),"The development of mnemonic and mnemotechnical functions", In: R. W. Rieber (Ed.), *The History of the Development of Higher Mental Functions*, Plenum Press, New York, pp. 179-190

Watkins, D. A. and Biggs, J. B. (2001),"The paradox of the Chinese learner and beyond", In: D. A. Watkins and Biggs, J. B. (Eds.), *Teaching the Chinese Learner: Psychological and Pedagogical Perspectives*, Comparative Education Research Centre, Hong Kong, pp. 3-23

Whiteman, W. E. C. and Nygren, K. P. C. (2000), "Achieving the right balance: properly integrating mathematical software into engineering education", *Journal of Engineering Education*, **89**(3), pp. 331-336

Winer, B. J., Brown, D. R. and Michels, K. M. (1991),"Chapter 9: Latin squares and related designs", In: *Statistical Principles in Experimental Design*, McGraw-hill, Inc., New York,

Winston, W. L. (1994), *Operations Research - Applications and Algorithms*, Duxbury Press, Belmont, CA

Winston, W. L. (1996), "Issue in education: software decisions", *OR/MS Today*, **23**(5), http://lionhrtpub.com/ORMS/ORMS-10-96/education.html

Appendices

Appendix 1: Research Studies relating to conceptual tasks and CAS

Heid (1988)

Heid (1988)'s study compared three classes who were taking a first-year university calculus. Heid taught two of the classes using the black-box software, CAS. The two classes were called ‘Experimental 1’ and ‘Experimental 2’ and had 17 and 18 students, respectively. In Experimental 1, students were shown basic algorithms and computations whilst learning with the CAS, whilst in Experimental 2 the algorithms and computations were taught to the students at the end of the semester. The third class called the ‘Comparison’ had 122 students and was taught by another teacher using traditional teaching methods. This last class focussed on algorithms, that is,procedural knowledge. Heid taught with the CAS and made use of multiple representations by showing both algebraic and graphical representations. The quizzes and mid-term tests were constructed by Heid to include tasks that required the use of conceptual knowledge. Students using CAS were found to do better on these tasks (although she did not perform any statistical tests). The final examination was not created by Heid, and the questions were related to procedural knowledge. The students performed at a similar level in the examination across all three groups.

O'Callaghan (1998)

O'Callaghan (1998) conducted a study similar to Heid in which three university algebra classes took part. The first class had 42 students and was taught using the black-box software, CAS. The second and third classes were taught by traditional methods, and each had 32 students. The first and second classes were taught by O'Callaghan and the third class by another teacher. The students who used the CAS were taught in a manner that was expected to improve conceptual knowledge, such as the use of multiple representations. O'Callaghan found that the CAS students performed significantly better in tasks requiring conceptual knowledge ($F = 5.77$, $p \leq 0.01$) compared with the other classes who were taught procedural knowledge (algorithmic procedures). However, the traditional classes did significantly better in a pen-and-paper final examination containing mostly procedural tasks. O'Callaghan suggested that, as the students using the CAS had been significantly poorer in mathematics at the beginning of the semester, there was not sufficient evidence to suggest that the CAS students made less progress.

Palmiter (1991)

The study by Palmiter (1991) is quite similar to that of Heid, in that students in a university calculus course were assigned to two groups, one taught with the black-box software, CAS, the other taught by traditional methods (emphasising procedural knowledge). The classes were taught by different teachers. In the CAS group, students completed the course in 5 weeks whilst those in the traditional group completed the course in 10 weeks. Those in the CAS group dispensed with learning the computational aspect of calculus (as this was performed on the CAS) and concentrated on tackling conceptual tasks. Both groups at the end of their course were given identical tests with procedural and conceptual tasks. The students in the CAS group were allowed to use the CAS during the test. A significant difference was found between the classes for the conceptual and procedural tasks, with the CAS class performing better in both sets of tasks. As the CAS group computed their answers, they were not subject to algorithmic or arithmetic errors, which is probably why they scored higher in the procedural tasks. Palmiter was cautious over the interpretation of these results, since the traditional group took the test after a ten-week period and were required to perform procedural tasks by hand.

Kendal and Stacey (1999)

Kendal and Stacey (1999) observed three teachers teaching a calculus class at secondary school using CAS. The aim of the course was to use CAS to enhance conceptual understanding. However, the teachers used different methods to teach mathematics, something that Kendal and Stacey called 'teacher privileging'. They found that students who had been taught by teachers who privileged the use of conceptual knowledge performed better on the conceptual parts. All the students performed at a similar level on the procedural tasks, regardless of the teacher, even though some students solved the tasks using pen-and-paper whereas others used a CAS. One teacher privileged far less use of CAS (Teacher B) than another (Teacher C) but both of these teachers' students performed at a similar level on the conceptual tasks.

This result implied that it is how the students are taught with CAS rather than simply the use of CAS alone that can encourage conceptual thinking, This can be seen in Teacher A, who privileged CAS highly but whose students were still unable to do well on the conceptual tasks. This was perhaps because the students with Teacher A had directed teaching which favoured only one mathematical representation (algebraic). According to Prosser and Trigwell (1999) (p.158), directed teaching (or information transmission) is linked to surface learning by students. Teacher B used a guided discovery style (where leading questions were asked in the context of the task) which favoured deep learning, but this teacher also only favoured one representation (algebraic). Teacher C also used guided discovery (deep learning) but favoured all three representations (algebraic, graphical and linking algebraic and graphical). Thus, it seemed that a combination of multiple representations and guided discovery promoted the best conceptual knowledge.

Appendix 2: Print version of the Mathematics-Computing Attitudes and ALMQ Inventory

Students' attitudes towards mathematics, computers and studying

This questionnaire is intended to find out the attitudes of students towards studying mathematics with software in various courses and their approach to studying these courses. This survey forms part of my doctoral research on students learning mathematics with software.

Your answers are CONFIDENTIAL, and will not be divulged to anyone teaching this course. Thank you for your co-operation.

If you have any further questions or queries, you can contact me (Anesa Hosein) via email: A.Hosein@open.ac.uk or telephone 01908 659866.

Instructions: This questionnaire is divided into two parts. The first part you have to answer questions related to your attitudes towards the mathematics and computers for the course we emailed you about (check email for details). The second part looks at your studying approach for the mathematics component in this course.

Before you begin these two parts, we would like you to answer which of the following you consider to be your main discipline or area of study (tick only one):

( ) Mathematics/ Statistics

( ) Physical Sciences (e.g. Physics, Chemistry)

( ) Computer Science

( ) Engineering/ Technology

( ) Life Sciences (e.g. Biology, Agriculture, Environmental Science)

( ) Arts (e.g. Languages, History)

( ) Business Studies

( ) Economics

( ) Education

( ) Social Sciences

( ) Other (Please State): _____________

( ) Not Sure

Part 1:
Mathematics-Computing Attitude Inventory

Instructions: This questionnaire has a number of questions which seeks to find your attitudes towards software and mathematics. This questionnaire should be answered with respect to the software (check email for details) used in your course and the mathematics component.

For each item there is a row of numbers (1 – 5) corresponding to a five point scale. Please circle **one of the five numbers. The numbers stand for the following responses:**

1: I strongly agree with this statement

2: I agree with this statement

3: I neither agree/ disagree to this statement

4: I disagree with this statement

5: I strongly disagree with this statement.

Circle the appropriate number for each of the statements that you think that best applies to you. There is no right answer. The most important thing is to answer the question as honestly as possibly and give the answer that best describes you.

1. I prefer to work on my own than in a group.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

2. If I can avoid using a computer I will.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

3. I don’t understand how some people can get so involved with computers.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

4. The challenge of understanding mathematics does not appeal to me.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

5. I rarely review the material soon after a computer session is finished.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

6. If a computer procedure I am using goes wrong, I panic.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

7. I find it helpful to test understanding by attempting exercises and problems.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

8. I feel more confidence of my answers with a computer to help me.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

9. I find it helpful to make notes in addition to copying material from the computer screen, or obtaining a printout.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

10. Using a computer makes learning more enjoyable.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

11. By looking after messy calculations, computers make it easier to learn essential ideas.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

12. Having to spend a lot of time on mathematics problems frustrates me.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

13. I will work at a computer for long periods of time to successfully complete a task.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

14. I like to stick at a mathematics problem until I get it out.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

15. The way computers force you to follow a procedure annoys me.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

16. Having to learn difficult topics in mathematics does not worry me.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

17. When studying mathematics I try to link new ideas to knowledge I already have.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

18. I don’t understand how some people can get so enthusiastic about doing mathematics.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

19. I have a lot of confidence when it comes to mathematics.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

20. I find working through examples less effective than memorising given material.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

21. Computers help me to link knowledge e.g. the shapes of graphs and their equations.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

22. I find it difficult to transfer understanding from a computer screen to my head.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

23. I feel nervous when I have to learn new procedures on a computer.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

24. I don't usually make time to check my own working to find to correct errors.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

25. When learning new mathematics material I make notes to help me understand and remember.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

26. Using computers makes me mentally lazy.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

27. I like to revise topics all at once rather than space out my study.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

28. I prefer to work with symbols (algebra) than with pictures (diagrams and graphs).
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

29. I don't trust myself to get the right answers using a computer.
Strongly Agree Neutral Strongly Disagree
1 2 3 4 5

30. I can get good results in mathematics.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

31. I have a lot of self-confidence in using computers.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

32. If something about mathematics puzzles me, I find myself thinking about it afterwards.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

33. I am more worried about mathematics than any other subject.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

34. Mathematics is a subject I enjoy doing.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

35. Following keyboard instructions takes my attention away from the mathematics.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

36. I am not naturally good at mathematics.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

37. Mathematics is a subject in which I get value for effort.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

38. As a male/female (*cross out that which does not apply*) I feel disadvantaged in having to use computers.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

39. I like the freedom to experiment that is provided by a computer.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

40. The prospect of having to learn new mathematics makes me nervous.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

41. I enjoy thinking up new ideas and examples to try out on a computer.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

42. If I make a mistake when using a computer I am usually able to work out what to do for myself.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

43. I am confident that I can master any computer procedure that needed for my course.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

44. I can become completely absorbed doing mathematics problems.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

45. When I read a computer screen, I tend to gloss over the details of the mathematics.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

46. If something about mathematics puzzles me, I would rather someone gives me the answer than have to work it out myself.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

47. No matter how much I study, mathematics is always difficult to me.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

48. Computers help me to learn better by providing many examples to work through.

Strongly Agree		Neutral		Strongly Disagree
1	2	3	4	5

Part 2:

Approaches to Learning Mathematics Questionnaire

On the following pages are a number of questions about your ways of studying the mathematics component in your course.

For each item there is a row of numbers (1 – 5) corresponding to a five point scale. Please circle **one of the five numbers. The numbers stand for the following responses:**

1: this statement was only *rarely* true of me

2: this statement was *sometimes* true of me

3: this statement was true of me about *half the time*

4: this statement was *frequently* true of me

5: this statement was *always* or *almost always* true of me

Please answer each item. Do not spend a long time on each item; your first reaction is probably the best one. Do not worry about projecting a good image.

	only rarely				almost always
1. I am concentrating on studying mathematics largely with a view to the job situation when I graduate rather than because of how much it interests me.	1	2	3	4	5
2. I find that studying mathematics gives me a feeling of deep personal satisfaction.	1	2	3	4	5
3. I think browsing around is a waste of time, so I only study seriously the mathematics that's given out in class or in the course outline.	1	2	3	4	5
4. While studying mathematics I think of real life situation in which the material that I am learning would be useful.	1	2	3	4	5
5. I am worried about how my performance in mathematics will affect my overall assessment.	1	2	3	4	5
6. While realising that mathematical ideas are forever changing and knowledge is increasing, I need to discover what is meaningful for me.	1	2	3	4	5
7. I learn some things in mathematics by rote, going over and over them until I know them by heart.	1	2	3	4	5
8. In reading new material in mathematics I find that	1	2	3	4	5

	only rarely				almost always
I'm continually reminded of material I already know, and see the latter in new light.					
9. Whether I like it or not, I can see that doing well in mathematics is a way for me to get a good result in first year.	1	2	3	4	5
10. I feel that mathematics becomes interesting once I become involved in studying it.	1	2	3	4	5
11. In studying mathematics I am focusing more on the examples than the theoretical material.	1	2	3	4	5
12. Before I am satisfied, I find that I have to do enough work on mathematics until I personally understand the material.	1	2	3	4	5
13. I worry that even if I work hard in mathematics the assessment might not reflect this.	1	2	3	4	5
14. I find that studying mathematics is as interesting as a good novel or movie.	1	2	3	4	5
15. I restrict my study of mathematics to what is specifically set, as I think it is unnecessary to do anything extra.	1	2	3	4	5
16. I try to relate what I have learned in mathematics to material in other subjects.	1	2	3	4	5
17. I think it's only worth studying the mathematics that I know will be examined.	1	2	3	4	5
18. I become increasingly absorbed in mathematics the more I do.	1	2	3	4	5
19. I learn mathematics best from materials/tutors which have carefully prepared notes and outline the major points neatly.	1	2	3	4	5
20. I find most aspects of mathematics interesting and spend extra time trying to obtain more information about them.	1	2	3	4	5
21. I almost resent having to study mathematics but feel that the end result will make it all worthwhile.	1	2	3	4	5

	only rarely				almost always
22. I spend a lot of my free time finding out more about interesting aspects of mathematics.	1	2	3	4	5
23. I find it best to accept the mathematical statements and ideas of my teacher(s) and question them only under special circumstances.	1	2	3	4	5
24. I believe strongly that my aim in studying mathematics is to understand it for my own satisfaction.	1	2	3	4	5
25. I am prepared to work hard in mathematics, because I feel it will contribute to my employment prospects.	1	2	3	4	5
26. Studying mathematics challenges my views on how the world works.	1	2	3	4	5
27. I am very aware that tutors know a lot more mathematics than I do, so I concentrate on what they say, rather than rely on my own judgement.	1	2	3	4	5
28. I try to relate new mathematics material, as I am reading it, to what I already know.	1	2	3	4	5

Would you like to provide any comments on your learning of mathematics with the software in your course in the box below? This could include any complaints, problems, things you find easy etc.

Appendix 3: Print version of the background questionnaire, instructional materials, pre-test, post-test for expected-values pilot study

<u>Demographic Questionnaire</u>

1. Age Category

    18 <25 [ ]25<35 [ ]35 to <45 [ ]45 to <55 [ ]

    55 to <65 [ ]≥ 65 []

2. Highest category attained in maths

    GCSE or less [ ]A-Levels [ ]University [ ]

3. Gender

    Female [ ]Male [ ]

4. Maths Confidence (Indicate on the scale of 1 to 10, where your maths confidence lies, where 1 = Poor and 10= Excellent)

    Poor

Poor									Excellent
1	2	3	4	5	6	7	8	9	10

5. Computer Confidence (Indicate on the scale of 1 to 10, where your computer confidence lies, where 1 = Poor and 10= Excellent)

Poor									Excellent
1	2	3	4	5	6	7	8	9	10

6. How confident are you in using Microsoft Excel? (Indicate on the scale of 1 to 10, where your Excel confidence lies, where 1 = Poor and 10= Excellent)

Poor									Excellent
1	2	3	4	5	6	7	8	9	10

7. Have you heard of the term ‘expected value’?

    Yes [ ]No [ ]

8. Have you ever calculated/ used ‘expected value’?

    Yes [ ]No [ ]

Pre-Test Materials

Pre-test

Simple Probability and maximization and minimization questions (pre-test: 10 mins and suggested marking scheme)

1. What is the probability that a card drawn at random from a deck of cards will be an ace? (simple probability) [1]
2. If a dice is rolled, what is the probability that the dice will have a value of four or more? (simple probability) [1]
3. What is the probability that when a pair of six-sided dice are thrown, the sum of the numbers equals 5?[2]
4. What is the probability that when a pair of six-sided dice are thrown, the sum of the numbers equals 12?[2]
5. If a coin is tossed twice, what is the probability that on the first toss the coin lands heads and on the second toss the coin lands tails?[2]
6. If a coin is tossed twice what is the probability that it will land either heads both times or tails both times?[2]
7. A box contains red marbles and blue marbles. One marble is drawn at random from the box (each marble in the box has an equal chance of being drawn). If it is **red**, you win $1. If it is **blue**, you win nothing. You can choose between 2 boxes. Box A contains 3 red marbles and 2 blue ones. Box B contains 30 red marbles and 20 blue ones. Which box gives you a better chance of winning?[1]

Introductory Materials

Games/ Lotteries

When I buy a game ticket for example the Lotto, there is usually a very low probability I could win the jackpot (£1,000,000) but I may have a higher probability of winning one of the smaller prizes (£5).

We can have a simpler game than the lotto, where in this game everyone can acquire a prize if they have a ticket. The probability of winning any prize in this game lies between 0% and 100%.

Let's say in this simple game, there are only two prizes (I'll call it Prize 1 and Prize 2). Prize 1 has a value of £40 but there is a 12% chance of winning it. Prize 2 has a value of £5 with an 88% chance of winning it.

I can find something called the expected value, which tells me what my 'average' prize could be, if I played this game often (let's say more than 100 times). The expected value of this game is calculated as follows:

$$(0.12 \times £40) + (0.88 \times £5) = £9.20$$

Practice Materials

(time recorded)

Application Sharing

Sign in into either MSN or Windows messenger using username: remoteobservation@hotmail.co.uk and password: Observation. I would contact you through this using the voiceconv@hotmail.co.uk user account, where we would be able to have a voice and video conversation.

Using the appsharing@hotmail.co.uk user account through which you will be able to look and use the software. When I contact you for application sharing and you click accept, you might be asked to install a piece of software or not, which is the Windows Application Sharing software, Netmeeting, called conf.exe. Click OK to install and Run the software. You can install this beforehand by going into Start>Run> and typing C:\Program Files\NetMeeting\conf.exe

Once we've started application sharing, you will see a screen pop up within which the excel spreadsheet is embedded and shown. You can maximize this screen holding the excel sheet if you want, but **do not move** the excel spreadsheet inside the embedded window (this for video recording purposes).

Once the session has started, I would ask you to click '*Control*' which is located at the top of the window, and to ask for control by clicking '*Take Control*', once you've been granted control you'll be able to use the Excel spreadsheet. When you're finished go back to the 'Control' button and click '*Release Control*'.

Software

In the Excel spreadsheet provided, there is a worksheet to help you find the expected values for each game and then find the best game. In the '*Home*' sheet, this is the sheet you see when you enter the programme and this is where you enter the data you have received: you have to enter the probabilities and the prizes for each of the game in the pink cells.

There are six buttons labelled on top: '*Black*', '*White*', '*Grey*', '*Clear All*', '*Scrap*' and '*Answer*'. The '*Black*', '*White*' and '*Grey*' button carries you to the sheet that will aid you in finding the best game. The '*Clear All*' button clears all the cells where you've entered data in that sheet and in the '*Black*', '*White*' and '*Grey*' sheets. When you click either '*Black*', '*White*' and '*Grey*' buttons, the data previously stored on that page is automatically erased. The '*Scrap*' sheet is a sheet where you can do any calculations or typing that you wish to do, whilst in the '*Answer*' sheet you enter the answers for the problems you would do in the post-test.

There is also an area in the 'Home' sheet which says current answers, this is the current answers that are calculated when using the '*Black*', '*White*' and '*Grey*' sheets.

When using:

Black-Box sheet: In this sheet, there are six buttons: '*Game 1*', '*Game 2*', '*Game 3*', '*Best*', '*Clear All*' and '*Home*'. The *Game 1-3* buttons calculates the expected values of each game, whilst '*Best*' is used for finding the best game. The '*Clear All*' button clears the cells where a value has been calculated and the '*Home*' button carries you back to the '*Home*' page.

When using:

Grey-Box sheet: In this sheet, there are six buttons: '*Game 1*', '*Game 2*', '*Game 3*', '*Best*', '*Clear All*' and '*Home*'. The *Game 1-3* buttons calculates the expected values of each game, whilst '*Best*' is used for finding the best game. The '*Clear All*' button clears the cells where a value has been calculated and the '*Home*' button carries you back to the '*Home*' page.

When using:

White-Box Sheet: In this sheet, there are three areas in grey, where you have to calculate the expected value for each game. Each of these grey areas, have four buttons: '*Clear*', '*1st Prize*', '*2nd Prize*', '*Expected Value*'. The '*Clear*' button is used to clear the data enter in those cells in that grey area. The '*1st Prize*' (or '*2nd Prize*') button is used to calculate the average money I would win of prize 1 (prize 2) if I played numerous times. The '*Expected Value*' button helps calculate the expected value of the game.

Outside of the grey region, there are three buttons: '*Clear All*', '*Best*' and '*Home*'. The '*Clear All*' button is used for clearing all data entered in that sheet. The '*Best*' button helps in finding the best game, whilst the '*Home*' button carried you back to the '*Home*' sheet.

When using:

Scrap sheet: This sheet is only linked to the Home page by the '*Home*' button. In this sheet you can do any calculations you want only within the boxed region.

When using:

Answer sheet: This sheet is only linked to the Home page by the '*Home*' button. Here you would input the answers for each of the questions in the post-test. For the practice Question, and Questions 1-7 there is a drop down menu to choose the answer. You can add any other comments you want next to this box particularly for Questions 4 and 6 which requires a comment.

Practice question for software:

For each excel sheet, practice doing the following problem and talk aloud and tell me what you are doing.

Which of the following games would I get the best expected value for?

Game 1:

1st prize: 20% probability of winning £90

2nd prize: 80% probability of winning £46

Game 2:

1st prize: 30% probability of winning £60

2nd prize: 70% probability of winning £40

Game 3:

1st prize: 10% probability of winning £100

2nd prize: 90% probability of winning £10

Post-Test Questions

Mechanical problem:

1. Which of the following games would I get the best expected value for?

Game 1:

1st prize: 29% probability of winning £93

2nd prize: 71% probability of winning £33

Game 2:

1st prize: Expected prize of £129

2nd prize: 86% probability of winning £33

Game 3:

1st prize: 9% probability of winning £159

2nd prize: 91% probability of winning £16[3]

2. Which of the following games would I get the best expected value for?

Game 1:

1st prize: 32% probability of winning £87

2nd prize: Expected prize of winning £36

Game 2:

1st prize: Expected prize £55

2nd prize: 37% probability of winning £75

Game 3:

1st prize: 46% probability of winning £68

2nd prize: 54% probability of winning £45[3]

3. Which of the following games would I get the best expected value for?

Game 1:

1st prize: 47% probability of winning £105

2nd prize: Expected prize of winning £58

Game 2:

1st prize: Expected prize £98

2nd prize: 37% probability of winning £129

Game 3:

1st prize: 78% probability of winning £68

2nd prize: Expected prize of winning £135[3]

<u>Interpretive problem:</u>

1. Which of the following games would I get the best expected value? *r* is an arbitrary probability. Give your reasoning.

Game 1:

1st prize: (*r*-30%) probability of winning £56

2nd prize: Expected prize of £25

Game 2:

1st prize: *r* probability of winning £55

2nd prize: Expected prize of £25

Game 3:

1st prize: (*r* + 10%) probability of winning £25

2nd prize: Expected prize £21[3]

2. Which of the following games would I get the best expected value if *r* takes on its highest possible value? *r* is an arbitrary probability.

Game 1:

1st prize: *r* probability of winning £100

2nd prize: Expected prize of £1000

Game 2:

1st prize: (r+ 10%) probability of winning £100

2nd prize: Expected prize of £500

Game 3:

1st prize: (r - 40%) probability of winning £1000

2nd prize: Expected prize £10[3]

3. Which of the following games would I get the lowest expected value? r is an arbitrary probability. Give your reasoning.

Game 1:

1st prize: (r) probability of winning £100

2nd prize: Expected prize of £50

Game 2:

1st prize: (r+20%) probability of winning £200

2nd prize: Expected prize of £100

Game 3:

1st prize: (r + 20%) probability of winning £100

2nd prize: Expected prize £50[3]

Constructive problem:

1. Which of the following games would I get the best expected value?

Game 1

Prize 1: 19% probability of winning £159

Prize 2: Expected prize of £33

Game 2

Prize 1: Expected prize of £28

Prize 2: 12% probability of winning another game which has a 61% probability of winning £50 for prize 1 or I can expect to win £170 for prize 2.

Game 3

Prize 1: 89% probability of winning £59

Prize 2: Expected prize of £185 [4]

2. Joan's assets consist of £10,000 in cash and a £90,000 home. During a given year, there is a 0.001 chance that Joan's home will be destroyed by fire or other causes. How much would Joan be willing to pay for an insurance policy that would replace her home if it was destroyed?[3]

3. My current income is £40,000. I believe that I owe £8,000 in taxes. For £500, I can hire a CPA to review my tax return; there is a 20% chance that she will save me £4000 in taxes. Should I hire the CPA?[3]

Appendix 4: Participants' profiles

Partici	Box	Seq.	Gender	Maths Level	Maths Conf.	Task Scores[2]	Degree
1	Black	1	Female	University	High (8) [1]	0.8	Phys. Sciences
2	Black	1	Male	A-level	High (8)	0.7	Phys. Sciences
3	Black	1	Male	A-level	High (9)	0.8	Life Sciences
4	Black	1	Female	A-level	High (8)	0.4	Life Sciences
5	Black	2	Male	University	High (8)	0.9	Phys. Sciences
6	Black	2	Male	GCSE	Low (5)	0.3	Life Sciences
7	Black	2	Male	A-level	High (7)	1.1	Life Sciences
8	Black	2	Female	A-level	High (8)	0.6	Life Sciences
9	Black	3	Female	A-level	High (7)	1.8	Life Sciences
10	Black	3	Female	University	Low (5)	0.4	Life Sciences
11	Black	3	Female	A-level	High (7)	0.8	Life Sciences
12	Black	3	Female	GCSE	Low (2)	0.8	Life Sciences
13	Glass	1	Male	University	High (8)	1.3	Phys. Sciences
14	Glass	1	Female	A-level	Low (6)	0.2	Other
15	Glass	1	Male	GCSE	Low (5)	0.8	Life Sciences
16	Glass	1	Male	A-level	High (7)	0.6	Phys. Sciences
17	Glass	2	Male	University	Low (6)	0.8	Phys. Sciences
18	Glass	2	Female	University	High (8)	0.8	Life Sciences
19	Glass	2	Female	A-level	Low (6)	0.7	Life Sciences
20	Glass	2	Female	University	Low (5)	0.3	Life Sciences
21	Glass	3	Female	A-level	Low (4)	0.7	Life Sciences
22	Glass	3	Male	GCSE	Low (5)	0.4	Life Sciences
24	Glass	3	Female	GCSE	Low (3)	0.7	Life Sciences
25	Open	1	Male	A-level	High (8)	0.8	Phys. Sciences
26	Open	1	Female	A-level	Low (5)	0.8	Life Sciences
27	Open	1	Female	A-level	High (8)	0.3	Life Sciences
28	Open	1	Male	A-level	High (7)	1.2	Life Sciences
29	Open	2	Female	University	High (7)	0.5	Phys. Sciences
30	Open	2	Male	GCSE	High (8)	1.0	Life Sciences
31	Open	2	Female	University	Low (5)	0.2	Other

Partici	Box	Seq.	Gender	Maths Level	Maths Conf.	Task Scores[2]	Degree
32	Open	2	Female	University	Low (5)	0.6	Other
33	Open	3	Male	A-level	Low (6)	1.0	Phys. Sciences
34	Open	3	Male	A-level	High (7)	0.6	Life Sciences
35	Open	3	Female	GCSE	Low (3)	0.3	Life Sciences
36	Open	3	Male	A-level	Low (5)	0.8	Life Sciences
37	Black	2	Male	University	Low (6)	0.6	Phys. Sciences
38	Glass	3	Female	A-level	Low (5)	0.9	Phys. Sciences
39	Glass	2	Male	University	High (8)	0.4	Phys. Sciences

[1] Mathematics Confidence scores

[2] Mean performance scores for the interpretive and constructive tasks (out of 2)

Appendix 5: Consent form, pre-test, instructional materials, post-test and ASI for main linear programming study

Remote Observation Study Consent Form

I am asking for your consent for me to observe you and record both video and voice of you and the actions you undertake whilst on the computer as part of my remote observation. This data will form part of my doctoral research at the Open University's Institute of Educational Technology, under the supervision of Dr James Aczel, Dr. Doug Clow, and Prof. John Richardson.

If you tell me of any aspect of our session which you wish to remain private, I will not divulge it to anyone else; and if you wish me to destroy any of the data that you provide, I will do so. Otherwise, in reporting my research I may also describe our session and use short quotations from your words, video and voice clips for academic purposes such as reports, presentations at seminars and conferences. Every effort will be made to conceal your identity if it can be done without diminishing the academic illustration.

My notes of the session, the full audio, video and software recordings will be held securely and accessed only by myself and a transcriber trusted to handle confidential material.

If you accept these conditions, can you type your name in full below and click the 'Submit' button.

Anesa Hosein
(A.Hosein@open.ac.uk)

Pre-Test (print version)

1. Find the value of x in: $x + 5 = 8$?
2. Find the value of y in: $2y + 4 = 16$
3. If $t = 2$, find the value of x: $3x + 2t = 13$
4. Solve for the value of y: $4y + 2 > 12$
5. If $x = 5$, solve for the value of y: $3x + 5y <10$
6. Find the values of x and y in the following 2 equations: $2x + y = 10$ and $2x + 3y = 22$
7. Find the values of t and y in the following 2 equations: $2t + 3y = 8$ and $t + y = 3$

Instructional Materials

Linear Programming

Background to Linear Programming

Linear programming (LP) is a mathematics topic and has no relation to computer programming. It is often used in agriculture, business and engineering for finding the best profit of producing products (often called maximizing profit) or finding the least cost of producing the products (often called minimizing the cost).

Linear programming is related to algebra and to simultaneous equations. In simultaneous equations we may have the following two equations:

$$2x + 3y = 13$$

$$x + y = 5$$

and are then asked to solve for x and y. Here we have two equations and two variables (2 variables: x and y). We note, there are equal number of equations and variables (two each) and we can solve this to show that x =2 and y=3.

In linear programming, there may be several variables; let's say 6 variables, and only 3 equations. Mathematical theory says in order for us to solve for the 6 variables we must have 6 equations, but this is impossible in this situation. What we thus do in linear programming, is that we make 3 variables equal to 0, and that allows us to solve the other 3 variables with the 3 equations.

Also in linear programming, some of the equations may be linear inequalities such as:

$$2x + 3y \leq 13$$

$$x + y \leq 5$$

If we solved these two equations using simultaneous equations, this will mean that $x \leq 2$ and $y \leq 3$. That is, x can take any value from $-\infty$ to 2 and similarly y lies between $-\infty$ to 3. Often, in linear programming, the variables represent products produced, thus a proviso is made that the variables must be ≥ 0, as we cannot produce negative products.

This means that in this case, the x variables can range from 0 to 2 or can be written as $0 \leq x \leq 2$ and similarly for y, we can write $0 \leq y \leq 3$. This provides us with some leeway when we are trying to solve the problems.

The linear programming problem

To understand a linear programming problem in context of agriculture, business and engineering, let's take a simple problem of a farmer how many acres she must plant in wheat or corn to maximize her profit.

Problem: Farmer Jane wants to plant her land with wheat or corn. Each acre of land planted with wheat yields £200 profit; each with corn yields £300 profit. The labour and fertilizer used for each acre are given in the table below. One hundred workers and 120 tons of fertilizer are available. Farmer Jane wants to know how many acres of each crop she has to grow in order to have the best profit.

	Wheat	Corn
Labour/acre	3 workers	2 workers
Fertilizer/acre	2 tons	4 tons

To start with in a linear programming problem, there is always a clear objective such as maximizing profit or minimizing cost. In this case, Farmer Jane wants to maximize her profit. Therefore, we can say:

Let w = no. of acres of wheat to grow

c = no. of acres of corn to grow

z = Farmer Jane's profit

We can then set up an overall equation to represent Farmer Jane's objective

$$\text{Max } z = 200w + 300c \text{ (eq. 1)}$$

where 'Max z' represents that we want to maximize her profit (z). This equation just says for every acre of wheat Jane plants she will get £200 and for every acre of corn she plants she will get £300.

This equation representing her cost or profit is often called the **objective function**. Therefore, as we want to maximize Farmer Jane's profit we must find the highest values that w and c can take. However, we only have limited amount of labour and fertilizer, that is, we are constrained.

We can write the equation of the labour required in terms of acres of corn and wheat as follows:

$$3w + 2c \leq 100 \text{ (no. of labourers)(eq. 2)}$$

This equation means that every acre of wheat we grow requires 3 workers, every acre of corn we grow, we require 2 workers and that our total number of workers cannot exceed 100 (as there is only 100 labourers available). This type of inequality is called a **constraint**, and we can refer to this as our labourer constraint.

Our second limiting factor or constraint, is that of the fertilizer. We can write this as:

$$2w + 4c \leq 120 \text{ (Fertilizer constraint)(eq. 3)}$$

The fertilizer constraint says that for every acre of wheat we plant we need 2 tons of fertilizer, for every acre of corn we need 4 tons of fertilizer and we only have 120 tons available.

We can put all three equations (eq. 1 to 3) together, to formulate the whole linear programming problem:

Max $z = 200w + 300c$

Subject to (s.t.)

$3w + 2c \leq 100$ (Labourer Constraint)

$2w + 4c \leq 120$ (Fertilizer Constraint)

Two more inequalities can be included to indicate that both w and c are ≥ 0, but this is often understood to be true.

Setting up a Linear Programming Problem to Solve It

To solve the linear programming problem, we often employ a method called the simplex method. In order to do the simplex method, we must change all the inequalities (such as eq. 2 and 3) to equalities (that is,have an equal sign).

The objective function (eq. 1) is already an equality, but we can rewrite it to have all the variables on one side:

$$z - 200w - 300c = 0 \text{(eq. 4)}$$

Let's now look at the labourer constraint (eq. 2),

$3w + 2c \leq 100$ (Labourer Constraint)

We can rewrite this as

$3w + 2c + s_1 = 100$ (Labourer Constraint)(eq. 5)

where s_1 is called the slack variable. s_1 is added to ensure that the equations always add up to 100. For example, if $w = 10$ acres of wheat and $c = 20$ acres of corn, then for the labourer constraint (eq. 5):

$3(10) + 2(20) + s_1 = 100$

$30 + 40 + s_1 = 100$

$s_1 = 30$

Therefore, s_1 means there are 30 remaining labourers. Thus s_1, takes up the remaining labourers or the slack which is unaccounted for.

We can write a similar equation for the fertilizer constraint (eq. 3)

$2w + 4c \leq 120$ (Fertilizer Constraint)

We can rewrite this with a slack variable, s_2:

$2w + 4c + s_2 = 120$(eq. 6)

Therefore we can rewrite the three equations (eq. 4 to 6) to include all the variables as a list of equations to solve similarly to simultaneous equations:

Sequence 0 : $z - 200w - 300c + 0s_1 + 0s_2 = 0$

s.t.

Sequence 1:$0z + 3w + 2c + s_1 + 0s_2 = 100$

Sequence 2:$0z + 2w + 4c + + 0s_1 + s_2 = 120$(eq. 7)

Sequence 0, Sequence 1 and Sequence 2, are used to identify which equation we are referring to in the linear programming problem. Note that we have 5 variables but only 3 equations.

Solving a Linear Programming Problem

Now as we only have 5 variables and 3 equations, we must set (or make) two variables equal to 0. Now since we want to find the profit (represented by z), we should never set $z = 0$. However, we note that the simplest solution to these three equations, is if we set $w = 0$ and $c = 0$. Therefore, a solution to the problem can be found:

If we make

$w = 0$ and $c = 0$

then we can calculate z, s_1 and s_2 as

$z = 0$ from Sequence 0; $s_1 = 100$ from Sequence 1; $s_2 = 120$ from Sequence 2

As we were able to calculate the values of z, s_1 and s_2, these are termed the **basic variables (**BV**)** and they were easily calculated because they only occur once in all of the three rows. This solution means that we don't plant any corn or wheat, hence our profit, $z = 0$, and we have 100 labourers and 120 tons of fertilizer remaining. But, Farmer Jane needs to grow either some wheat and/or corn to receive a profit.

This is where we begin to employ the simplex method. The simplex method is basically a method where we try out different values of the variables to increase the value of z (the profit) but at the same time ensuring that we keep within the limits of the constraints. In the simplex method, we only try one variable at a time, the variable that we are trying is called the **pivot variable** (PV). For example, if we look at the objective function (eq. 1 or eq. 4), we'll see that the profit (z) will increase the most if we planted more acres of corn, since corn has the largest profit. Thus, we would try increasing the variable c (acres of corn) first, and thus c will be our first pivot variable. Currently, $c = 0$ and $w = 0$.

If we only look at number of labourers we have (eq.5), according to this equation, the largest no. of acres of corn we can grow has to be 50 (every acre of corn requires 2 labourers and we only have 100 labourers). Similarly according to the fertilizer constraint (eq. 6), we can only grow 30 acres of corn (every acre of corn requires 4 tons of fertilizer and we only have 120 tons of fertilizer). Thus, if we have to choose a value of c to satisfy both the fertilizer and labourer constraint, this has to be $c=30$ (if $c=50$, then we would have required 200 tons of fertilizer which we don't have!). Since the fertilizer constraint limited the value of c, this constraint is called the **pivot row** and corresponds to Sequence 2 of the linear programming problem (eq. 7).

As we are making $c = 30$, we thus have to set one of our basic variables: z, s_1 or s_2 to equal to 0, since we can only calculate the values of three variables. Since, it is Sequence 2 (the pivot row) that constrained the value of c, and it was this row that s_2, was calculated in, we can make $s_2 = 0$ and then be able to calculate c in this row.

Thus if we use $c = 30$ and have set $w = 0$ and $s_2 = 0$,

We have

$z = 9000$ from Sequence 0 and $s_1 = 40$ from Sequence 1.

As noted before, the reason why the basic variables values were able to be calculated easily was because they occurred only once in all the rows. If we did not know the value of c and would like to easily calculate it we would have to make it a basic variable. As at present, c occurs in three rows (Sequence 0, Sequence 1 and Sequence 2 of eq. 7), we would like it only to appear in one row, the pivot row. To remove it from the other rows, we perform something called elementary row operations (eros) which essentially means you multiply the pivot row by a number and then add it to another row, which by addition makes the value of c equal to zero in the other row. After doing this, we then restart the process again (called an **iteration**), with the new rows of equations created to determine which variable must increase in order for the profit to increase.

The best solution would be to plant 20 acres of corn and 20 acres of wheat which will give a profit of £10,000.

Software Instructions

There are 4 spreadsheets in the Excel Spreadsheet you'll be working with. Each spreadsheet is related to the question, i.e. the 1st spreadsheet is the practice question, the 2nd the 1st problem, the 3rd spreadsheet is the 2nd problem etc. For each problem in the post-test you will use the corresponding spreadsheet.

On each spreadsheet there are 5 buttons.

Input Problem: Click this button to enter the problem that you wish to solve. You can change the values/coefficients of the variables here. If you've change the variables, the problem starts back at the beginning. You may want to click **Reset** to remove any previous solution.

Iteration: Click this button to solve the problem, until a pop-up tells you the problem has been solved

Reset: Click this button to remove the solution found, i.e. to start all over again.

Answer Form: Click this button when you want to enter your answer for the problem

Clear All: Click this button when you want to start afresh. It erases everything in the sheet (Be careful!!)

Practice Question

You would use the following question to practice with the software.

Max $z = 200w + 300c$

s.t.

$3w + 2c \leq 100$ (Labourer Constraint)

$2w + 4c \leq 120$ (Fertilizer Constraint)

Post-Test

Problem 1

a) Solve

Max $z = 2x + y$

s.t.

$2x + y \leq 100$ (constraint A)

$x + y \leq 80$ (constraint B)

$x \leq 40$(constraint C)

b) Now, x refers to the no. of toy trains manufactured and y refers to the no. of toy soldiers manufactured whilst constraint A refers to painting hours and constraint B to carpentry hours. Interpret what this solution means to the toy company who wants to maximize their profit by producing toy trains and toy soldiers. Provide as detailed an answer as possible.

c) If the profit per train has increased by £1, how would this affect the number of toy trains and toy soldiers being sold and why? Provide as detailed an answer as possible.

Problem 2

a) Solve

$$\text{Max } z = 30x + 15y + 10t$$

s.t.

$$8x + 6y + 2t \leq 48 \text{ (Constraint A)}$$

$$8x + 4y + 3t \leq 40 \text{(Constraint B)}$$

$$4x + 3y + t \leq 16 \text{ (Constraint C)}$$

$$y \leq 5 \text{(Constraint D)}$$

b) Let x= no. of desks manufactured, y = number of chairs manufactured and t = number of stools produced. Let also Constraint A = number of hours available for carpentry/day (i.e. building the product), Constraint B = feet of lumber available, Constraint C = the number of hours/day available for finishing (i.e. painting and polishing the product) and D is the demand for the number of chairs. Which product(s) was not produced and give the possible reason(s) why? Give as detailed an answer as possible.

c) If the number of hours available for carpentry/day is increased from 48 to 60 hours, how would this change what the Furniture Company manufactured and why? Give as detailed an answer as possible.

Problem 3

a) Solve

$$\text{Max } z = 6x + 8y + 13t - u$$

s.t.

$$3x + 4y + 6t - u \leq 0 \text{ (Constraint A)}$$

$$2x + 2y + 5t \leq 100 \text{(Constraint B)}$$

$$u \leq 90 \text{(Constraint C)}$$

b) Why do we allow linear programming to have ≤ constraints rather than just < constraints? Which variable will we not want to have a high value for? Give as detailed an answer(s) as possible.

c) If u can be made greater than 90, what is the largest value that it can be? And why that value? Give as detailed an answer as possible.

Approaches to Study Inventory (Text Version)

This questionnaire intends to find the way your approach to studying for this exercise. There are 10 statements. For each statement, please select on the scale, the one that best describes your approach to studying during this exercise. There are no right answers, so, answer what best described your attitude.

5 - means that you definitely agree

4 - means that you agree, but with reservations

3 - means that you really find it impossible to give a definite answer

2 - means that you disagree, but with reservations

1 - means that you definitely disagree

1) I often had trouble making sense of the things I had to learn.
Definitely Agree 5 4 3 2 1Definitely Disagree

2) I set out to understand for myself the meaning of what I had to learn.
Definitely Agree 5 4 3 2 1Definitely Disagree

3) Much of what I learned seemed no more than lots of unrelated bits and pieces in my mind.
Definitely Agree 5 4 3 2 1Definitely Disagree

4) In making sense of the new ideas, I often related them to practical or real-life contexts.
Definitely Agree 5 4 3 2 1Definitely Disagree

5) Ideas I came across in the exercise often set me off on long chains of thought.
Definitely Agree 5 4 3 2 1Definitely Disagree

6) I looked at the evidence carefully to reach my own conclusion about what I was learning.
Definitely Agree 5 4 3 2 1Definitely Disagree

7) It was important for me to follow the argument or to see the reasons behind things.
Definitely Agree 5 4 3 2 1Definitely Disagree

8) I tended to take what I had been given at face value in the exercise without questioning it much.
Definitely Agree 5 4 3 2 1Definitely Disagree

9) In reading the instructional materials, I tried to find out for myself exactly what the author meant.
Definitely Agree 5 4 3 2 1Definitely Disagree

10) I just went through the motions of the exercise without any interest where it would lead.
Definitely Agree 5 4 3 2 1Definitely Disagree

Appendix 6: Marking scheme

Problem 1

Mechanical: z=100, x =40 and y =20[1 mrk]
For Open-Box students: There is an alternative solution: z=100, x=20 and y=60 [1 mrk]

Interpretive: The company has to produce 40 trains and 20 soldiers to get a maximum profit of £100 [1 mrk].
As constraints 1 and 3 are binding, it means that the painting hours will be used up and as well the company would meet the demand of 40 trains. There are 20 hrs of carpentry hours remaining [1 mrk].

For Open-Box students: The company has to produce 20 trains and 60 toy soldiers to get a maximum of £100 [1 mrk].
As constraints 1 and 2 are binding it means that the painting and carpentry hours will be used up and the company falls short of 20 toy trains for the demand [1 mrk]

Constructive: The profit increase should not affect the number of x and y produced as x is constrained by the demand; however, the overall profit would increase by £40, thus profit would be £140 [2 mrk] (1mrk for the number of toys would not increase and 1 mrk for the reason)

For OB students:
The profit increase may affect the number of x produced, since although the painting and carpentry are binding, y can be decreased and x can be increased, since there are still 20 toy trains in demand. If x increases to 40, this would be possible since from Constraints 1 and 2, this would mean the maximum that y can be is 20, if x is a maximum of 40. This would mean the profit would be £140. [1 mrk for x would increase and 1 mrk for the profit].

Problem 2
Mechanical: z =140, x =2, y=0 and t=8 [1 mrk]

<u>Interpretive:</u> *y* was not produced because it was not profitable. [1 mrk] *y* is not as profitable as *t* because more resources is needed to produce *y* that is about 3 times the resources needed by *t* and only produces \$5 more profit [1mrk]. <u>Constructive:</u> It would not make a difference as we already have enough carpentry hours. [2 mrk] (1 mrk that the production wouldn't increase and 1 mrk for that there weren't enough carpentry hours)
Problem 3 <u>Mechanical:</u> z=105, x=0, y=0, t=15, u=90 [1 mrk] <u>Interpretive</u>: If we let it be < then the values can approach let's say 90 but could never reach 90. However, with ≤ this allows the constraint to have any leeway from 0 to 90. [1mrk] We should not want a high value of *u*, since it is negative and hence reduces the profit. [1 mrk] … ½ marks is allowed if only *u* is answered. <u>Constructive</u>: *u* can be made as high as 200, as this would make *y* to be mostly manufactured and it needs 200 *u* to make it equal (constraint A). [2 mrk] (1 mrk that *u* can be made for 200 and 1 mrk as to why it can only be made as high as 200)

Appendix 7: Statistical annexes: explanations, tables and detailed findings

Annex 1: ANOVA assumptions and violations

Normality, Homogeneity of Variance and Effect Sizes

In some of the within-subjects variables of Problems and Tasks, heterogeneous variance was found based on Levene's test of homogeneity. Howell (2002) indicates that if the largest variance is 'no more than four times the smallest' then the ANOVA should be valid (p.340-341). However, he also cautioned that unequal variances and unequal sample sizes should not be mixed since it made the ANOVA less robust. Alternatively, Howell (2002) (p. 342) suggested using the Welch Procedure. Welch's procedure required using the statistic:

$$w_k = \frac{n_k}{s_k^2}$$

where w_k was the calculated Welch statistic, n_k was the sample number in the k^{th} treatment and s_k was the standard deviation within the k^{th} treatment. However, examining each within-subjects variable in this study, when they were sub-divided into their Boxes and Mathematics Confidence groups, showed that for those variables in which Levene's test indicated that there were a significant difference in variation, that these variables had a group which had a variance (or standard deviation) of zero. This usually resulted because the mean was zero (that is,everyone within that group failed to score). This meant that if the Welch Procedure was used that often w_k would not have a value since it will be dividing by zero and hence not a viable option.

Glass *et al.* (1972) alternatively provided guidelines when performing an ANOVA with heterogeneity of variance and unequal sample sizes. They indicated that:

- "Actual α exceeds nominal α when smaller samples are drawn from more variable populations"

- "Actual α is less than nominal α when smaller samples are drawn from less variable populations" (p.273).

The nominal alpha, α, they referred to is the level of probability that the null hypothesis is rejected at, normally 0.05. As small samples were used in the present study from a variable population, then perhaps the first condition applied to this study.

Further, Schultz (1985) indicated that the Levene's test tended to be "unduly conservative for small, odd sample sizes ($n \leq 7$)" (p.456). The group sizes used in this study were quite small as noted previously in Table 22. Therefore, it was likely that Levene's test was conservative when the variance for some groups was zero which resulted in the test being significant, but this was perhaps expected as this thesis investigated how students perform depending on which software they were using and how this was affected by Mathematics Confidence. Thus, the ANOVA seemed reasonably robust for the analysis used in this study and the results of the ANOVAs were accepted.

Further, when effects and interaction were found to be significant, this did not necessarily mean that the differences between the groups were large. To measure this difference, a measure of effect size was used. According to Olejnik and Algina (2003), effect size is defined as:

> a standardized index and estimates a parameter that is independent of sample size and quantifies the magnitude of the difference between populations or the relationship between explanatory and response variables. (p.434)

Cohen (1988) suggested criteria for 'small', 'medium' and 'large' effects for different measures of effect size. In ANOVAs, effect sizes can be measured using either f^2 or η^2 (eta squared) which are related by the following formula:

$$\eta^2 = {f^2}\Big/{1+f^2}\,.$$

The effect sizes for ANOVAs are presented in Table 50

Table 50: Effect sizes for η^2 for ANOVAs

	ANOVAs	
	f^2	η^2
Small	0.01	0.01
Medium	0.06	0.06
Large	0.16	0.14

The effect size that was used in this study was the partial eta squared (η_p^2). There are no set criteria for η_p^2. Olejnik and Algina (2003) indicated that η_p^2 provided estimates of effect size that might be larger that the actual effect size if a blocking variable (for example gender and perhaps in this study mathematics confidence) was used when compared with a study where there were no blocking variable used. Olejnik and Algina (2003) noted that one may argue that the blocking variable provides a stronger research design and hence a larger effect size.

However, according to Tabachnick and Fidell (2007) η^2 is flawed when there are a number of other independent variables, as in a one-way design η^2 will be larger if a two-way design was used with two independent variables (including the previous variable). They further explained that the reason for this is that the interaction increases the size of the total variance particularly if one of the effects is large. Thus they indicated that η_p^2 will be better since it is a proportion of the "variance attributable to the effect of interest plus error" (p.55).

However, Ramanau (2007) reported from his personal communication with Richardson that in Cohen's book effect sizes are applied to one-way analysis of covariance (ANCOVA). In this case, Cohen partialled out (i.e. removed) the effects due to the covariates, but then used the same criteria of small, medium and large effects. Richardson continued that on the same basis the partial effect sizes obtained in ANOVAs with multiple variables could be taken to represent effect sizes. He explained the reason why this is because "the effects of other independent variables and the interactions with those variables are partialled out before computing the proportion of variance explained" (p.166) which is an analogous procedure to the one-way ANCOVA. Thus the guidelines used for η_p^2 are the same as those shown for η^2 above (i.e. small, medium and large effects are 0.01, 0.06 and 0.14, respectively). Thus, in this research the guidance for effect size used was that for η^2 but with the caution that the value of η_p^2 might in principle be larger than that value of η^2, and that the values of η_p^2 across all the effects and interactions in a research design will not add up to 1.

Further, when an interaction was significant, Fisher's Least Significant Difference (LSD) was used for post-hoc analysis. The LSD takes into account the confidence limits to determine where there were any differences. When using LSD, there should be an overall significant F as this protected the overall family-wise error rate from being greater than 0.05 (Howell, 2002: p. 391). Howell (2002), however, cautioned that the LSD should be used only if there were three means involved since this ensured that the family-wise error rate stayed at 0.05. Since there were three or fewer means being compared at any time (i.e. Boxes, Tasks and Mathematics Confidence), the LSD procedure can be used.

Annex 2: Reason for using mathematics confidence as a split variable.

As Mathematics Confidence could influence how students perform, this was included as a covariate in the original ANOVA and an Analysis of Covariance (ANCOVA) had to be performed. However, the deviation scores for mathematics confidence were used in the ANCOVA since Van Breukelen and Van Dijk (2007) indicated that in SPSS, the covariate adjusts the grand mean and to ensure that the grand mean is not influenced then the covariates must have a mean of zero. As Sequence did not influence the scores, this variable was dropped from the repeated measures ANCOVA. By adding the Mathematics Confidence as a covariate (see Table 51), it was found that the Task × Box interaction was now significant ($F(2,34) = 5.09$, $p = 0.01$).

Table 51: ANCOVA for software Box, Mathematics Confidence, Problem and Task

	SS	df	MS	F	p	η_p^2
Between Subjects						
MathsConf	0.94	1	0.94	1.41	0.24	0.04
Box	0.28	2	0.14	0.21	0.81	0.01
SS within	22.77	34	0.67			
Within Subjects						
Problem	12.91	2	6.46	28.15	0.00	0.45
Problem × MathsConf	0.93	2	0.47	2.03	0.14	0.06
Problem × Box	1.04	4	0.26	1.14	0.35	0.06
Problem × SS within	15.60	68	0.23			
Task	17.88	1	17.88	75.28	0.00	0.69
Task × MathsConf	1.10	1	1.10	4.64	0.04	0.12
Task × Box	2.42	2	1.21	5.09	0.01	0.23
Task × SS within	8.08	34	0.24			
Problem × Task	4.34	2	2.17	7.59	0.00	0.18
Problem × Task × MathsConf	0.64	2	0.32	1.12	0.33	0.03
Problem × Task × Box	1.48	4	0.37	1.29	0.28	0.07
Problem × Task × SS within	19.45	68	0.29			

According to Winer, Brown and Michels (1991), a covariate can only adjust the between subjects effect but not a within-subject effects. In this case, the adding of the Mathematics Confidence covariate adjusted the significance of the Task × Box interaction, where Task is a within-subject. Gilmore (2007) indicated that this was a problem that SPSS had in that it did not present Winer *et al.* (1991)'s ANCOVA. Further, to acquire Winer's ANCOVA, two ANCOVAs had to be done on SPSS one with and without the covariates. This matter was handled in SPSS Resolution No. 22133 (SPSS-Knowledgebase, 2008). However, Anstey *et al.* (2007) explained that the

ANCOVA presented by SPSS was better than Winer *et al.* (1991)'s original ANCOVA, since Winer *et al.* (1991)'s was incomplete. They explained that the simplest case using Winer's model would be to have one within-subject effect (T) and one between-subject effect (G). Thus, they continued that the Winer's ANCOVA model in notational form would be:

$$y_{ij} = \mu + \alpha x_i + \beta G_i + \tau T_j + \gamma (G \times T)_{ij} + \varepsilon_{ij}$$

where y_{ij} referred to the outcome for individual *i* measured for factor T_j, G_i was the treatment received by individual *i*, μ was the grand mean, ε_{ij} was the error and x_i was the covariate value of individual *i* and which was constant across the within-subject effect, T. The letters α, β, τ and γ are the coefficients of the variables. Further, they indicated that since G and *x* shared the same subscript, then the covariate would only adjust the between-subject effect but not the within-subjects effect.

However, Anstey *et al.* (2007) indicated that another type of ANCOVAs can be made by including the covariate in the within-subject effect which they explained is the default ANCOVA that SPSS produces. Their equation for this ANCOVA was as follows:

$$y_{ij} = \mu + \alpha x_i + \beta G_i + \tau T_j + \lambda (x \times T)_{ij} + \gamma (G \times T)_{ij} + \varepsilon_{ij}$$

They indicated that the interaction term ($x \times T$) did not adjust the within-subject effect of T, but it did adjust the interaction term ($G \times T$) since they both shared the same subscript. This was perhaps why in this study the ANCOVA indicated that the Box × Task interaction had become significant. Anstey *et al.* (2007) further explained that if there was a significant λ (the coefficient representing the gradient of the $x \times T$ scores), this implied there was a violation of the assumption of homogeneity of slopes which was usually tested beforehand when carrying out a Winer's ANCOVA. In this case, the

Task × MathsConf interaction was significant ($F(1,34) = 4.64$, $p = 0.04$), where MathsConf variable represented Mathematics Confidence.

D'Alonzo (2004) explained that when there is heterogeneity of the regression in an ANCOVA, there were generally two methods to deal with this assumption within a repeated measures design. The first method required using lengthy calculations such as the method proposed by Delaney and Maxwell (1983) which involved picking points and doing simultaneous inferential procedures to determine exactly the effect of the covariate. Alternatively, the covariate can be split into a low or high category and be used as a between subject effect (D'Alonzo, 2004; Owen and Froman, 2005). Although Owen and Froman (2005) thought that reducing the covariate into two categories should be avoided, they however felt that within a repeated measures ANCOVA design this was perhaps the most legitimate case for doing this. The advantage of having the mathematics confidence split into two categories was that an ANOVA can be performed and this meant that any variance due to the interaction of the Box and Mathematics Confidence can now be determined.

Annex 3: ANOVA for determining Sequence and Question effects

Factors/ Effects	SS	df	MS	F	p
Between Subjects					
Sequence	0.60	2	0.30	0.40	0.67
Box	0.85	2	0.42	0.57	0.57
Sequence × Box	1.17	4	0.29	0.39	0.81
SS within groups	21.72	29	0.75		
Within Subjects					
Problem	12.43	2	6.22	24.40	0.00
Question	0.69	2	0.35	1.31	0.28
Question × Box	0.15	4	0.04	0.14	0.96
Problem × Box	0.70	4	0.18	0.69	0.60
(Question × Problem)'	0.21	2	0.10	0.39	0.68
(Question × Problem × Box)'	1.09	4	0.27	1.04	0.40
Question × SS within groups	15.28	58	0.26		
Task	17.27	1	17.27	60.75	0.00
Task × Sequence	0.03	2	0.01	0.05	0.95
Task × Box	1.54	2	0.77	2.71	0.08
Task × Sequence × Box	1.24	4	0.31	1.09	0.38
Task× SS within groups	8.24	29	0.28		
Question × Task	0.71	2	0.36	1.20	0.31
Problem × Task	4.24	2	2.12	7.29	0.00
Question × Task × Box	1.43	4	0.36	1.20	0.32
Problem × Task × Box	1.61	4	0.40	1.38	0.25
(Question × Problem × Task)'	0.02	2	0.01	0.03	0.97
(Question×Problem×Task×Box)'	0.46	4	0.12	0.39	0.82
Question×Task×SS within groups	17.30	58	0.30		

Annex 4: ANOVA for the difference between interpretive and constructive scores for each problem

	SS	df	MS	F	P	η_p^2
Between Subjects						
Box	5.68	2	2.84	6.51	0.00	0.29
MathConfRec	3.99	1	3.99	9.15	0.01	0.22
Box × MathConfRec	0.46	2	0.23	0.53	0.60	0.03
SS within	13.96	32	0.44			
Within Subjects						
ProbDiff						
ProbDiff × Box	7.17	2	3.58	6.70	0.00	0.17
ProbDiff × MathConfRec	3.17	4	0.79	1.48	0.22	0.09
ProbDiff × Box × MathConfRec	2.65	2	1.33	2.48	0.09	0.07
ProbDiff × SS within	3.38	4	0.85	1.58	0.19	0.09

Annex 5: Detailed quantitative findings for performance scores

Test	Performance Scores	*p* –value
1.Problem	Problem 2 (1.04) > Problem 1 (0.48) Problem 2 (1.04) > Problem 3 (0.55) Problem 1 (0.48) ≈ Problem 3 (0.55)	$p < 0.01$
2.Tasks	Interpretive (0.98) > Constructive (0.40)	$p < 0.01$
3.Maths Confidence	Higher MC (0.79) > Lower MC (0.58)	$p = 0.07$
Problem and Tasks 4.Interpretive 5.Constructive	 Problem 2 (1.23) ≈ Problem 3 (1.03) Problem 2 (1.23) > Problem 1 (0.68) Problem 3 (1.03) ≈ Problem 1 (0.68) Problem 2 (0.84) > Problem 1 (0.29) Problem 2 (0.84) > Problem 3 (0.07) Problem 1 (0.29) > Problem 3 (0.07)	$p < 0.01$
Problem and Maths Confidence 6.	 Problem 1: Higher MC (0.64) > Lower MC (0.32) Problem 2: Higher MC (1.00) ≈ Lower MC (1.08) Problem 3: Higher MC (0.75) > Lower MC (0.35)	$p < 0.01$
Tasks and Maths Confidence 7.	 Interpretive: Higher MC (1.18) > Lower MC (0.70) Constructive: Higher MC (0.41) ≈ Lower MC	$p < 0.01$

	(0.40)	
Boxes and Tasks		$p < 0.01$
8.	Interpretive: Glass-Box (1.09) ≳ Black-Box (0.83) Open-Box (1.01) ≈ Glass-Box (1.09) Open-Box (1.01) ≈ Black-Box (0.83) Constructive: Black-Box (0.58) ≈ Glass-Box (0.30) ≈ Open-Box (0.32) Difference between Interpretive and Constructive performance scores: Black-Box (0.25) < Glass-Box (0.79) Glass-Box (0.79) ≈ Open-Box (0.69) Black-Box (0.25) < Open-Box (0.69)	

Annex 6: Detailed quantitative findings for frequency of exploration

Exploration	Frequency of Exploration	p -value
9.Tasks	Constructive (33%) > Mechanical (5%) Constructive (33%) > Interpretive (2%)	$p < 0.01$
Constructive Task Exploration		
10.Problems	Problem 2 (61%) > Problem 1 (18%) Problem 2 (61%) > Problem 3 (21%)	$p < 0.01$
Problems and Maths Confidence		
11.	Problem 1: Higher MC (6 students) vs Lower MC (1 student) Problem 2 and Problem 3: Similar	$p = 0.09$
12.Boxes	Black-Box (44%) > Glass-Box (33%) > Open-Box (22%)	$p < 0.01$
13.Boxes and Maths Confidence	Higher MC: Black-Box (56%) > Glass-Box (17%) ≈ Open-Box (17%) Lower MC: Black-Box (17%) ≈ Glass-Box (41%) ≈ Open-Box (28%)	$p < 0.01$

Annex 7: Detailed quantitative findings for constructive exploration and performance scores

Constructive Exploration and Performance	Frequency of Scoring when Exploring Constructive Tasks	p -value
14.Scored	Explored (82%) > Not-Explored (9%)	$p < 0.01$
15.Problems	Problem 1: Explored (71%) > Not-Explored (13%) Problem 2: Explored (100%) > Not-Explored (20%) Problem 3: Explored (38%) > Not-Explored (0%)	$p < 0.01$
16.Boxes	Black-Box: Explored (88%) > Not-Explored (18%) Glass-Box: Explored (69%) > Not-Explored (4%) Open-Box: Explored (88%) > Not-Explored (7%)	$p < 0.01$
17.Maths Confidence	Explored: Higher MC (90%) ≈ Lower MC (72%)	$p > 0.1$

Annex 8: Detailed quantitative results for frequency of explanations

Explanations	Frequency of Explanations	*p* -value
18.Sequence	Sequence 2 (45%) > Sequence 1 (27%) Sequence 3 (39%) > Sequence 1 (27%) Sequence 2 (45%) ≈ Sequence 3 (39%)	$p = 0.05$
19.Problems	Problem 2 (42%) ≳ Problem 1 (30%) ≈ Problem 3 (33%)	$p = 0.08$
20.Problem and Explanation Type	Real-Life: Problem 1 (39%) > Problem 3 (8%) Problem 2 (50%) > Problem 3 (8%) Mathematical: Problem 3 (58%) > Problem 1 (21%) Problem 3 (58%) > Problem 2 (34%)	$p < 0.01$
21.Tasks	Constructive (41%) > Interpretive (29%)	$p < 0.01$
22.Tasks and Explanation Type	Real-Life: Interpretive (31%) ≈ Constructive (34%) Mathematical: Constructive (47%) > Interpretive (28%)	$p < 0.01$
23.Problems, Interpretive Task and Explanation Type	Real-Life: Problem 1 (66%) > Problem 2 (13%) ≈ Problem 3 (11%) Mathematical: Problem 3 (50%) > Problem 2 (21%) ≈ Problem 1 (13%)	$p < 0.01$

24.Problems, Constructive Task and Explanation Type	Real-Life: Problem 1 (68%) > Problem 2 (32%) > Problem 3 (5%) Mathematical: Problem 3 (66%) > Problem 2 (47%) > Problem 1 (29%)	$p < 0.01$
25.Boxes and Explanation Types	Real-Life: Glass-Box (41%) > Open-Box (24%) ≈ Black-Box (32%) Mathematical: Black-Box (42%) ≈ Glass-Box (31%) ≈ Open-Box (40%)	$p = 0.07$
26.Boxes, Real-Life Explanations and Maths Confidence	Lower MC: Black-Box (38%) ≈ Glass-Box (37%) ≈ Open-Box (31%) Higher MC: Glass-Box (50%) > Open-Box (17%) ≈ Black-Box (30%)	$p = 0.02$
27.Mathematical: Real-Life Explanations for Boxes	Ratio of Explanations Black-Box: 1.32 Glass- Box: 0.75 Open-Box: 1.71	

Annex 9: Detailed quantitative findings for explanations and performance

Explanations and Performance	Frequency of Scoring depending on Explanation	*p* -value
28.Problems and Explanations	Real-Life: Problem 1: Scored (17%) > Not-Scored (60%) Problem 2: Scored (49%) ≈ Not-Scored (54%) Problem 3: Scored (11%) ≈ Not-Scored (5%) Mathematical: Problem 1: Scored (25%) ≈ Not-Scored (18%) Problem 2: Scored (41%) > Not-Scored (0%) Problem 3: Scored (53%) ≈ Not-Scored (62%)	$p < 0.01$
29.Tasks and Explanations	Real-Life Interpretive: Scored (33%) ≈ Not-Scored (21%) Constructive: Scored (26%) ≈ Not-Scored (28%) Mathematical Interpretive: Scored (28%) ≈ Not-Scored (26%) Constructive: Scored (68%) > Not-Scored (37%)	$p < 0.01$

Annex 10: Detailed quantitative findings for processing level scores

Processing Levels	Processing Level Scores	*p-value*
30.Sequence	Surface Processing Level Sequence 2 (10.9) > Sequence 1 (6.5) Sequence 3 (8.9) ≈ Sequence 1 (6.5) Sequence 2 (10.9) ≈ Sequence 3 (8.9) Deep Processing Level Sequence 1 (23.7) ≈ Sequence 2 (24.8) ≈ Sequence 3 (24.6)	$p < 0.01$

Appendix 8: List of publications arising from this research

Hosein, A., Aczel, J., Clow, D., & Richardson, J. T. E. (2008), “Comparison of black-box, glass-box and open-box software for aiding conceptual understanding”, *Proceedings of the 32nd annual conference of the International Group for the Psychology of Mathematics Education (PME 32),* Vol. 3, Morelia, Mexico, pp. 185-192 (Peer-reviewed)

Hosein, A., Aczel, J., Clow, D., & Richardson, J. T. E. (2008), “Mathematical thinking of undergraduate students when using three types of software”, *Proceedings of the International Congress on Mathematics Education,* Monterrey, Mexico, http://tsg.icme11.org/document/get/531 (Peer-reviewed)

Hosein, A., Aczel, J., Clow, D., & Richardson, J. T. E. (2007), “An illustration of student's engagement with mathematical software using remote observation”, In Woo, J.-H., Lew, H.-C., 2. Park, K.-S. & Seo, D.-Y. (Eds.), *Proceedings of the 31st annual conference of the International Group for the Psychology of Mathematics Education (PME 31)*, Vol. 3, Seoul, Korea, pp. 49-56 (Peer reviewed)